Deep Learning

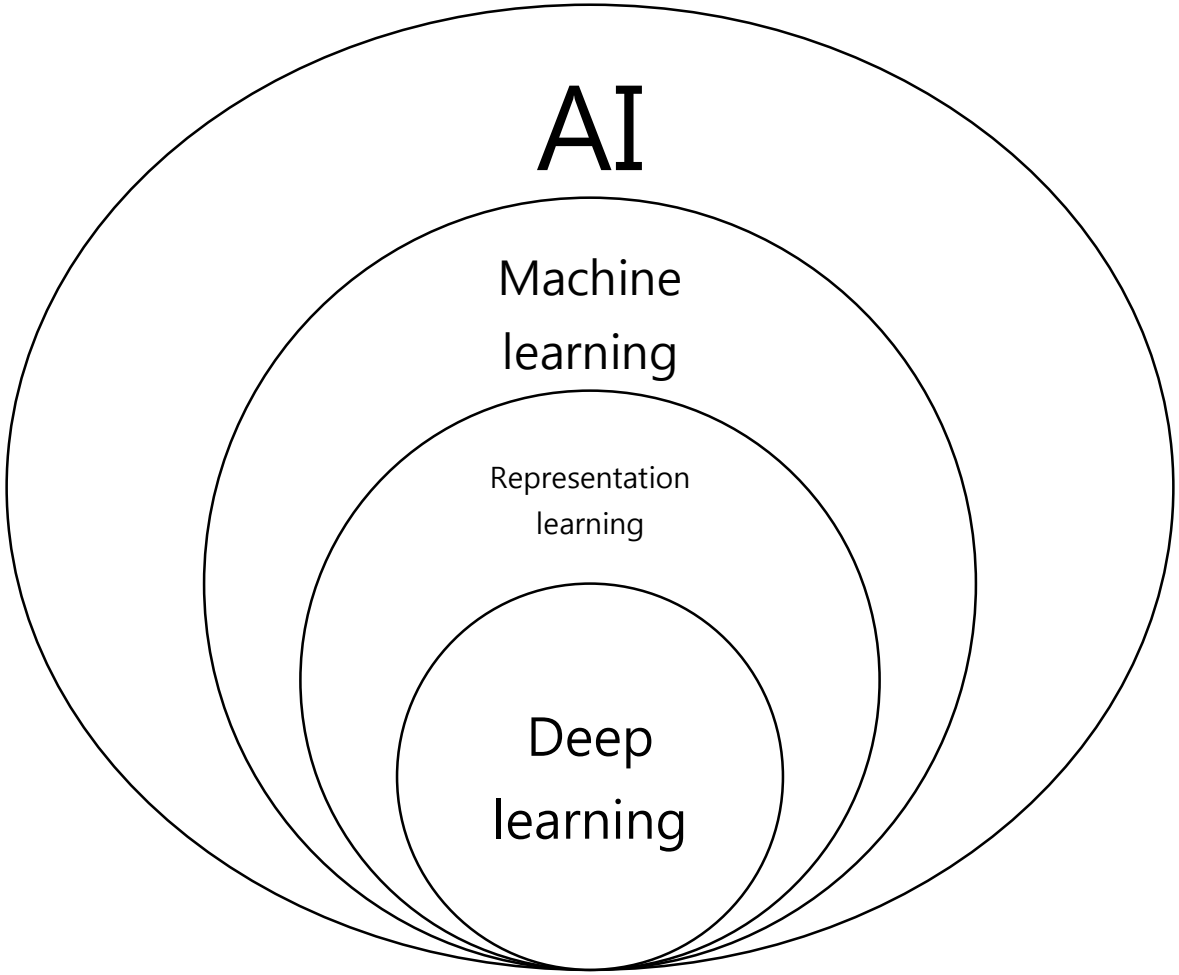
feat. Python

by 이성주

seongjoo@codebasic.io

@LeeSeongjoo

큰 그림


AI
Machine learning
Representation learning
Deep learning

(.ipynb)

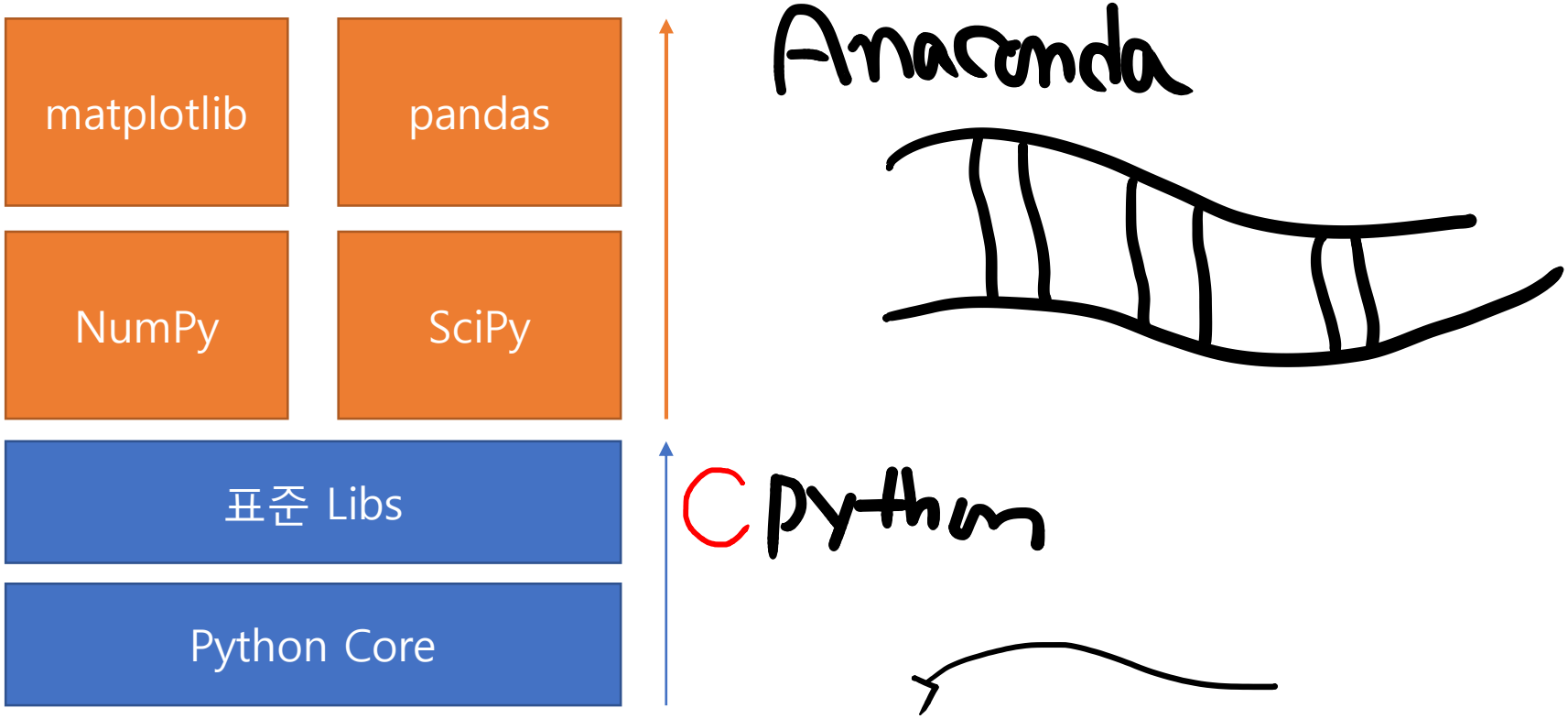
배포판

IPython

Jupyter

TensorFlow

$pip install pkg ...

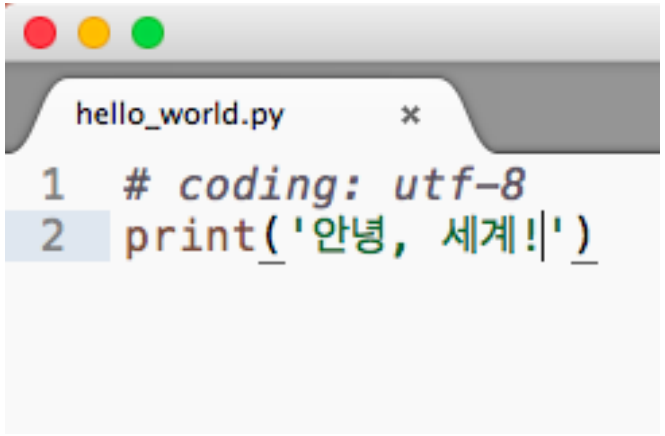


Hello, World!

$ python

>>> print('안녕, 세계!')

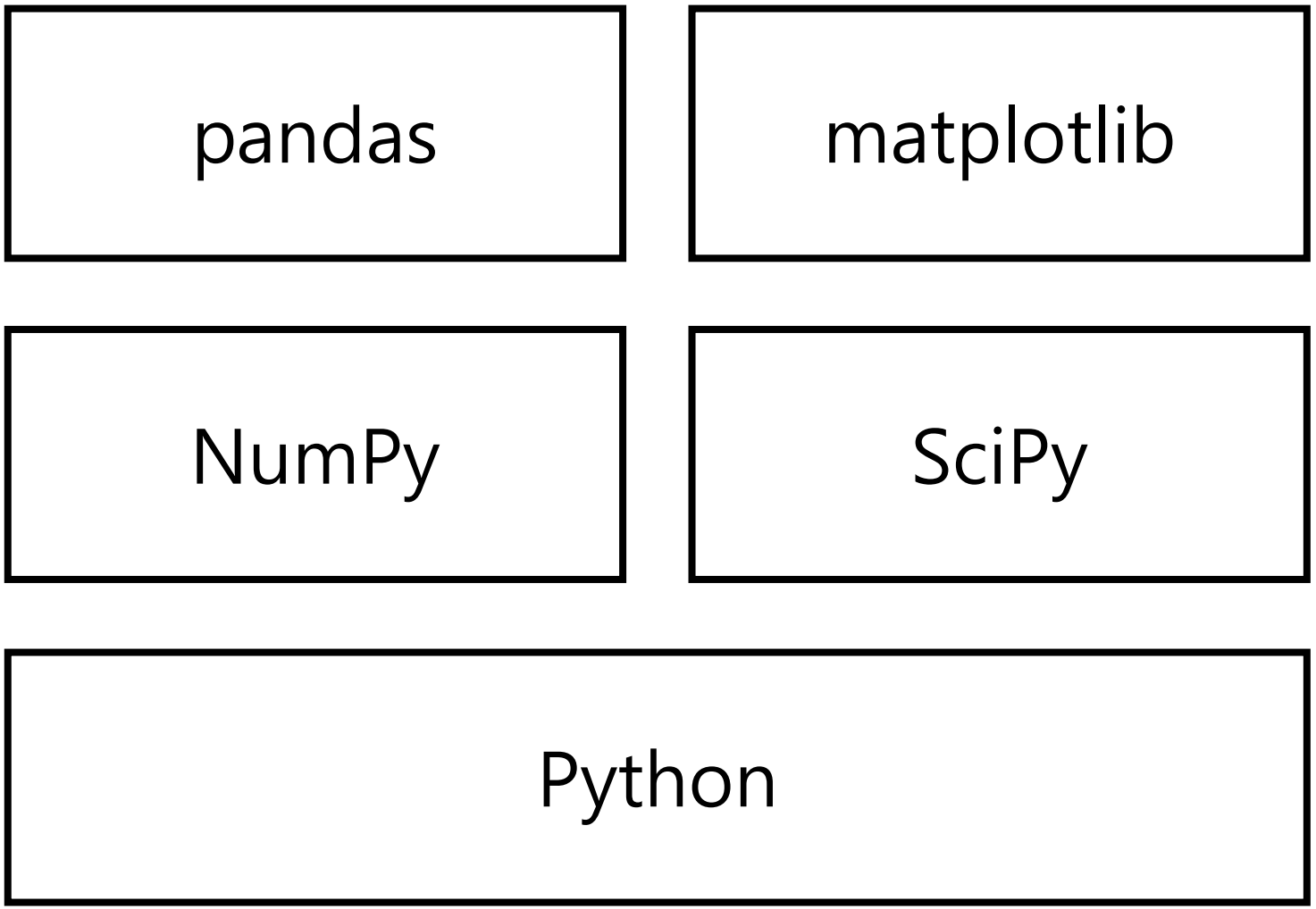
$ python hello_world.py



Jupyter Notebook

In [1]:
```
%matplotlib notebook

import pandas as pd
import numpy as np
import matplotlib

from matplotlib import pyplot as plt
import seaborn as sns

ts = pd.Series(np.random.randn(1000), index=pd.date_range('1/1/2000', periods=1000))
ts = ts.cumsum()

df = pd.DataFrame(np.random.randn(1000, 4), index=ts.index,
                  columns=['A', 'B', 'C', 'D'])
df = df.cumsum()
df.plot(); plt.legend(loc='best')
```

```
/opt/conda/lib/python3.5/site-packages/matplotlib/font_manager.py:273: UserWarning: Matplotlib is building the font c
ache using fc-list. This may take a moment.
  warnings.warn('Matplotlib is building the font cache using fc-list. This may take a moment.')
/opt/conda/lib/python3.5/site-packages/matplotlib/font_manager.py:273: UserWarning: Matplotlib is building the font c
ache using fc-list. This may take a moment.
  warnings.warn('Matplotlib is building the font cache using fc-list. This may take a moment.')
```


pandas
matplotlib
NumPy
SciPy
Python

NumPy

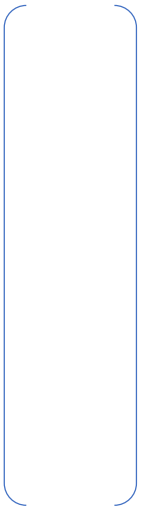
수치연산을 위한 라이브러리

텐서

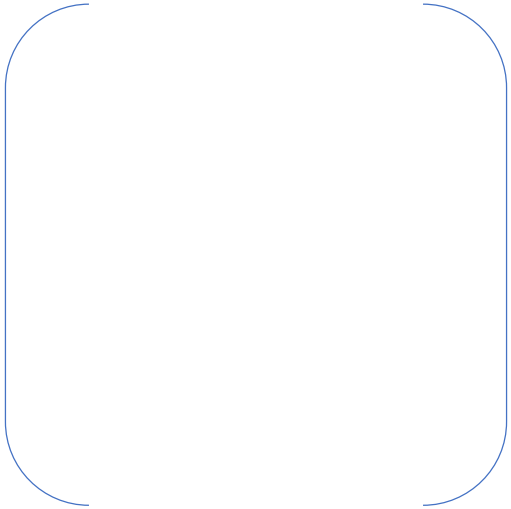
스칼라

x

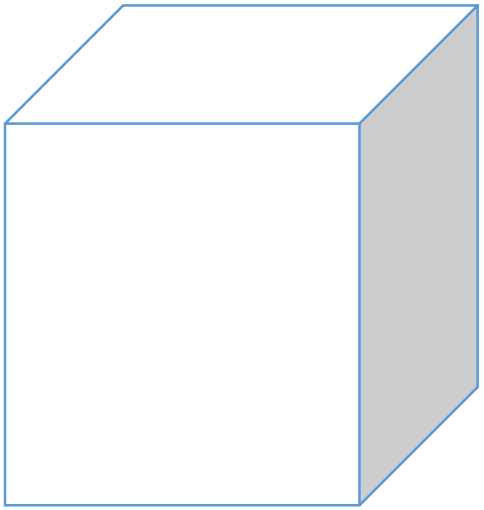
벡터



행렬



텐서 (n > 2)



dtype

PY3	NumPy
int	(u)int8/16/32/64
float	float16/32/64
⋮	⋮

dtype=int32

([1, 2, 3])

.astype(float)

([1.0, 2.0, 3.0])

Shape

$([1,2,3]) \rightarrow (3,)$

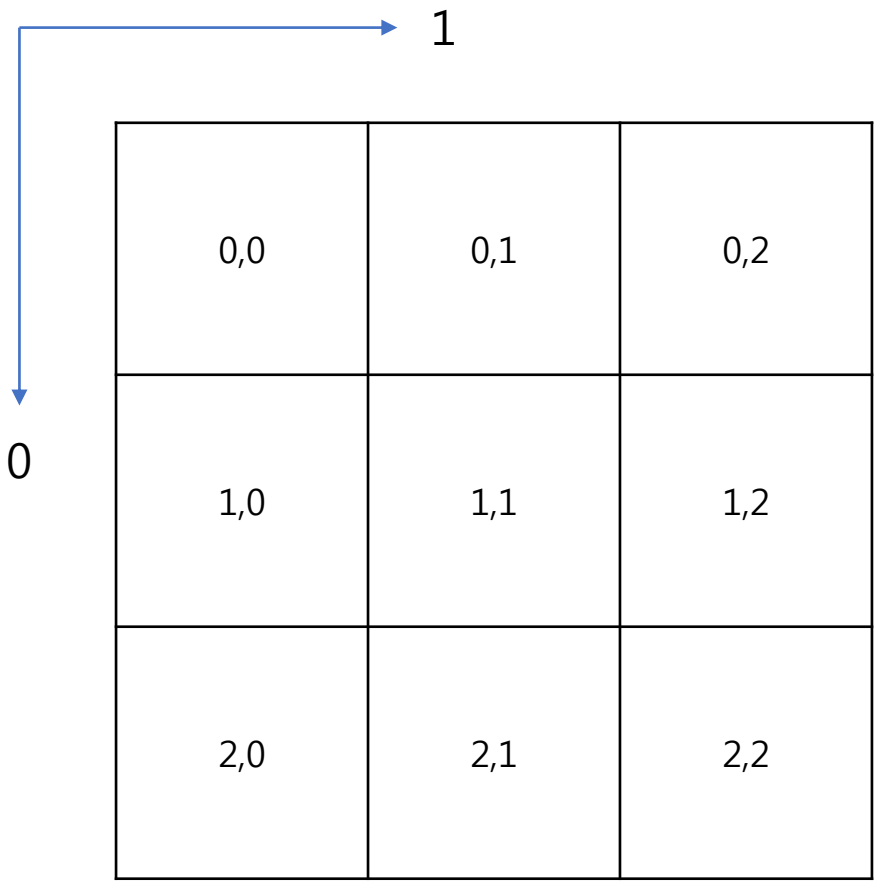
$\left(\begin{bmatrix}1,2\\3,4\end{bmatrix}\right) \rightarrow (2,2)$

```
([1,2,3,4]).reshape(2,2)
```

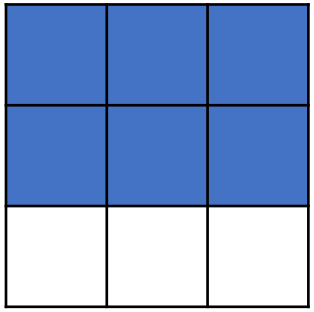
→

```
([[1,2],
  [3,4]])
```

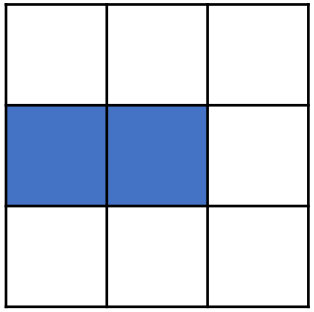
색인



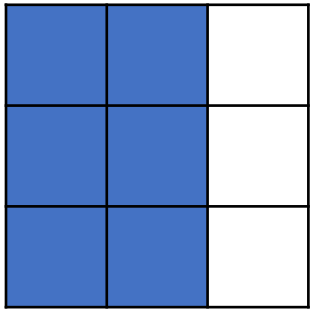
슬라이스



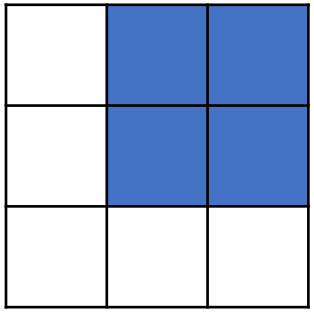
arr[0:2]
arr[:2]



arr[1, :2]



arr[:, 0:2]
arr[:, :2]



arr[:2, 1:]

팬시 색인

arr[[0, 2]]

0	1	2
6	7	8

arr[[2, 0]]

6	7	8
0	1	2

0	1	2
3	4	5
6	7	8

arr[:, [0, 2]]

0	2
3	5
6	8

arr[:, [2, 0]]

2	0
5	3
8	6

불리언 색인

원소별 논리연산

~x1	x1	x2	&	|	^
True	False	False	False	False	False
False	True	False	False	True	True
True	False	True	False	True	True
False	True	True	True	True	False

배열 전치

arr

0	3
1	4
2	5

(i, j)

arr.T

0	1	2
3	4	5

(j, i)

유니버설 함수

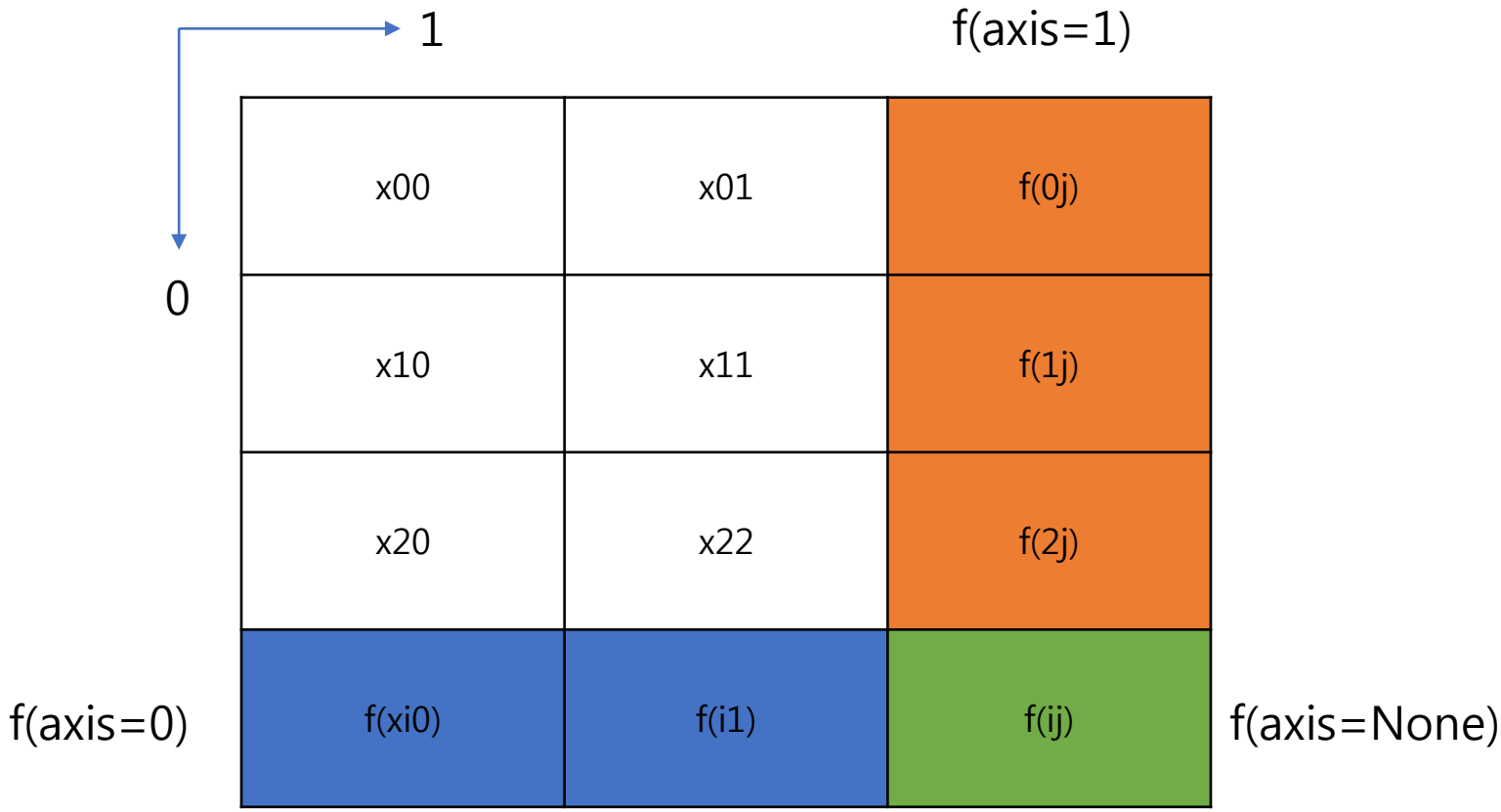
x	y	ufunc(x)	ufunc(x, y)
x1	y1	f(x1)	f(x1, y1)
x2	y2	f(x2)	f(x2, y2)
x3	y3	f(x3)	f(x3, y3)

배열 단위 조건 선택

np.where(cond, x, y)

cond	x	y	np.where
True	x1	y1	x1
False	x2	y2	y2

함수 연산 축


1
f(axis=1)
0
x00
x01
f(0j)
x10
x11
f(1j)
x20
x22
f(2j)
f(xi0)
f(i1)
f(ij)
f(axis=0)
f(axis=None)

집합 함수

함수	설명
np.unique(x)	고유값
np.intersect1d(x, y)	x, y 교집합
np.union1d(x, y)	x, y 합집합
np.in1d(x, y)	y에 해당하는 x
setdiff1d(x, y)	y에 해당하지 않는 x
setxor1d(x, y)	x, y 어느 한 쪽에만 있는 값

선형대수

x

x0	x1
x2	x3

y

y0	y1
y2	y3

x*y

x0y1	x1y1
x2y2	x3y3

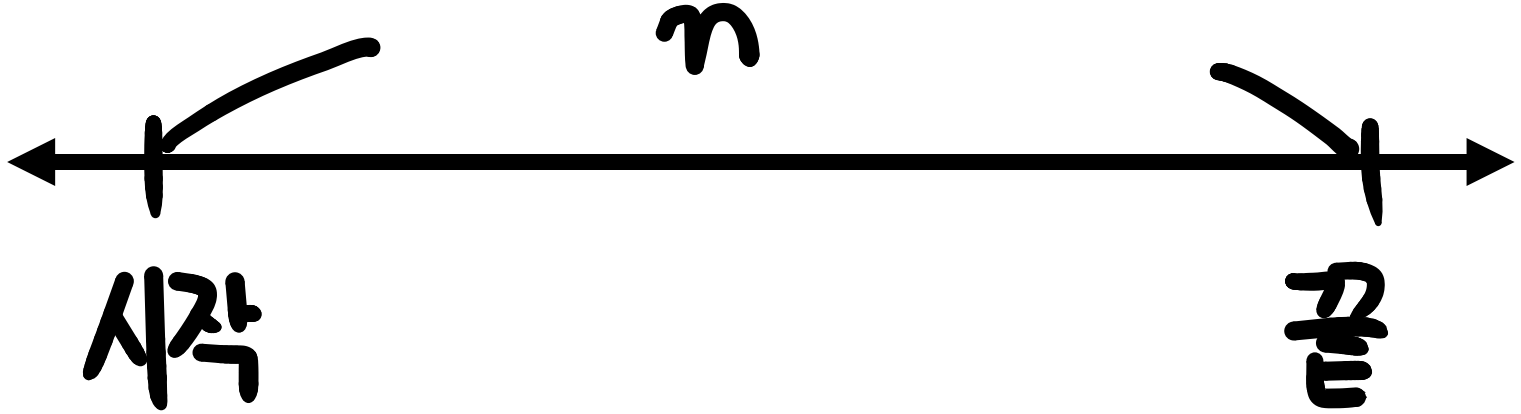
np.dot(x,y)

x0y0+x1y2	x0y1+x1y3
x2y0+x3y2	x2y1+x3y3

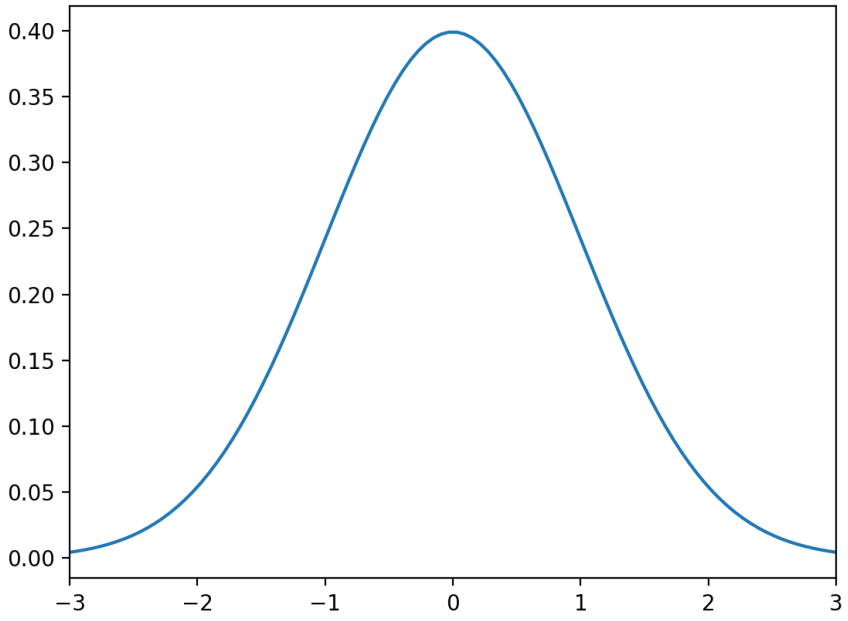
np.arange(시작, 끝, 간격)

≈ np.array(range(...))

np.linspace(시작, 끝, 구간개수)



난수

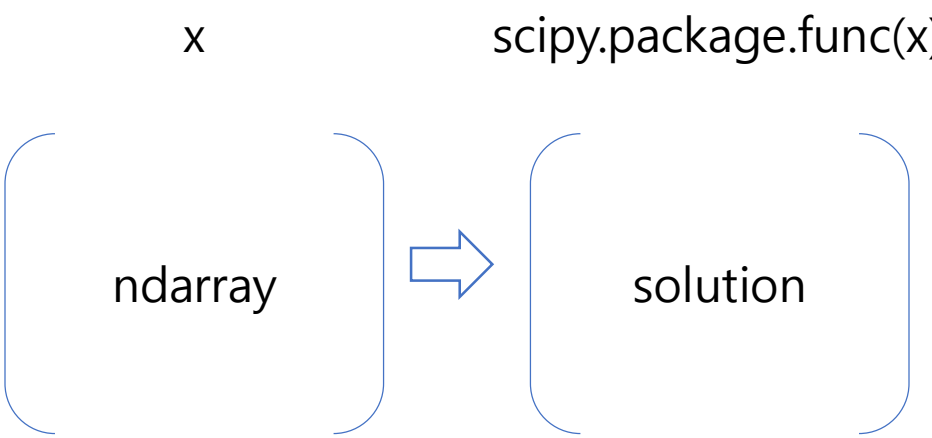


np.random	설명
seed	난수 발생 초기값
randint	[low, high) 범위 무작위 정수 생성
rand	균등분포(0, 1)
randn	정규분포(0, 1)
uniform	균등분포(low, high)
normal	정규분포(mean, std)
permutation	무작위로 섞인 배열 반환
shuffle	배열을 무작위로 섞는다

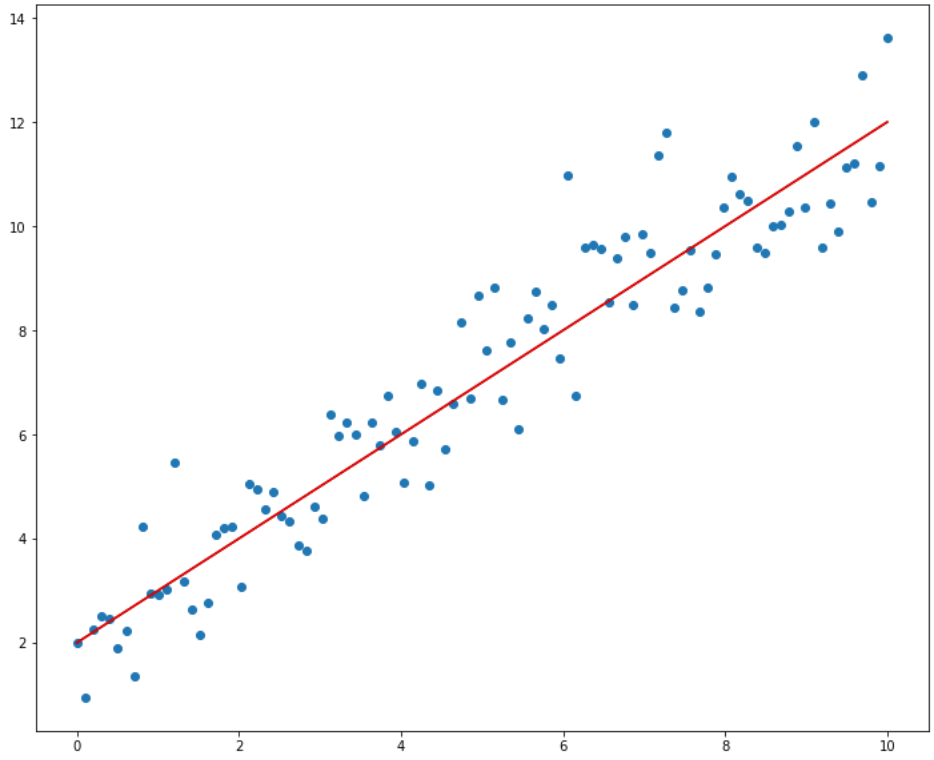
SciPy

과학계산용 함수 패키지

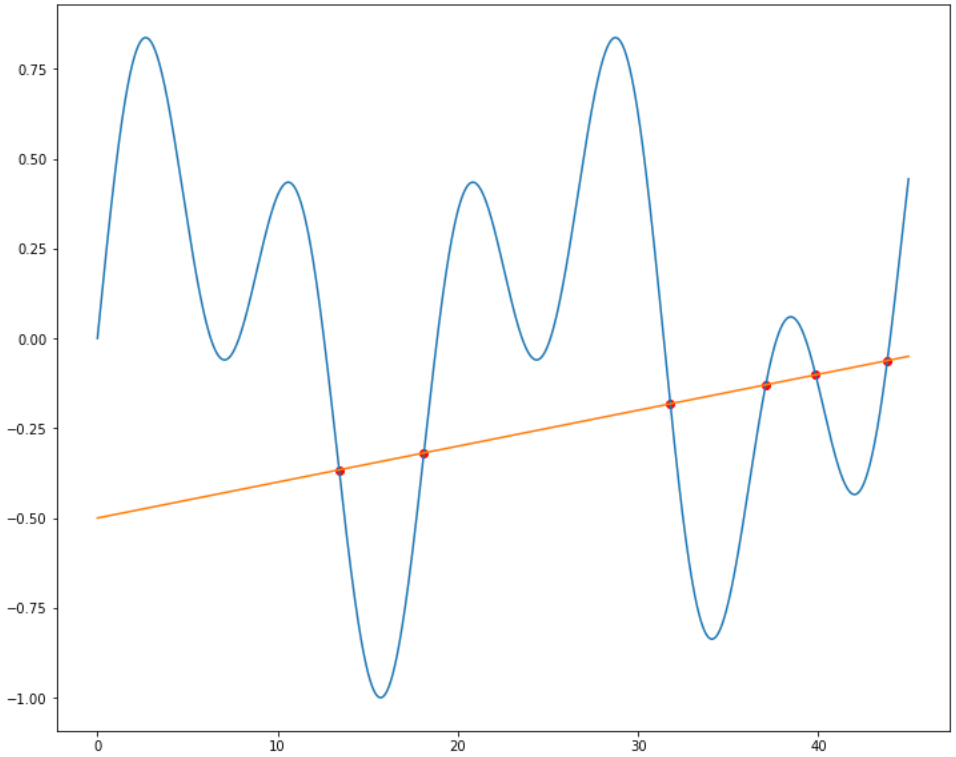
NumPy와 SciPy



데이터 피팅



해 구하기



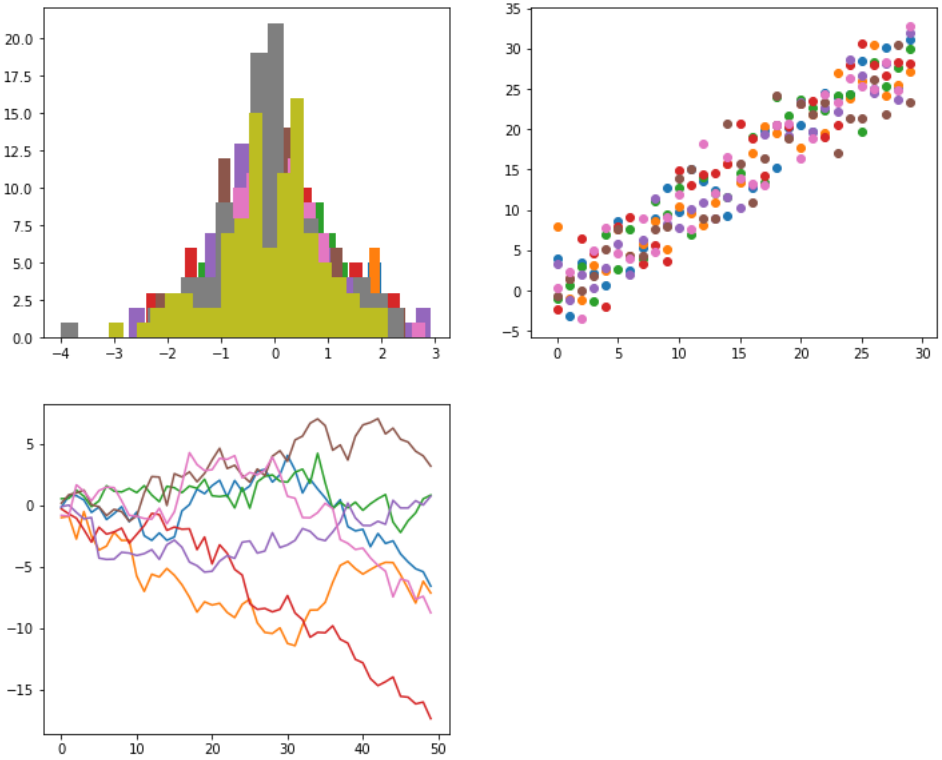
희소행렬

$$\begin{pmatrix} 1, 0, 0, \ldots 0, 0 \\ 0, 2, 0, \ldots 0, 0 \\ \\ 0, 0, \ldots 3, 0, 0 \\ \\ 0, 0, 0, \ldots 0, 0 \end{pmatrix}$$

matplotlib

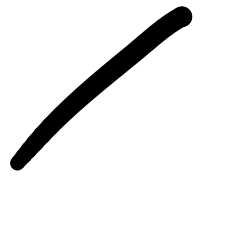
그래프 라이브러리

Figure


20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
−4
−3
−2
−1
0
1
2
3
35
30
25
20
15
10
5
0
−5
0
5
10
15
20
25
30
5
0
−5
−10
−15
0
10
20
30
40
50

plt. 그래프유형

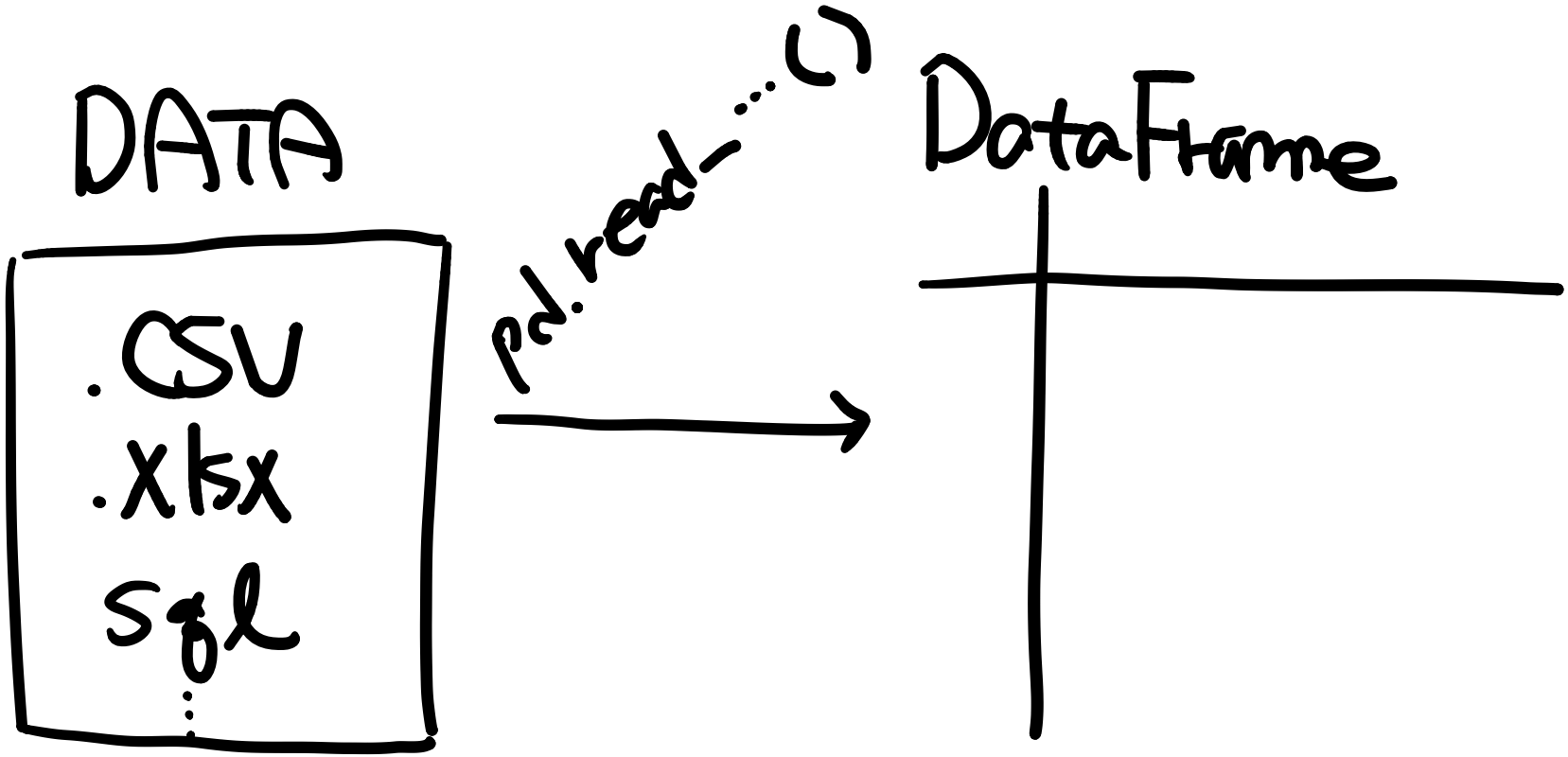
- .plot( )
- .scatter( )
- ⋮
- .imshow( )



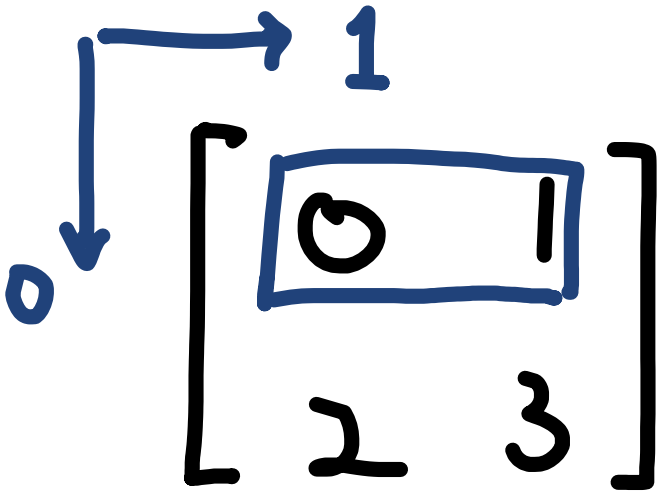
import pandas

pandas

Data I/O, 정리, 가공

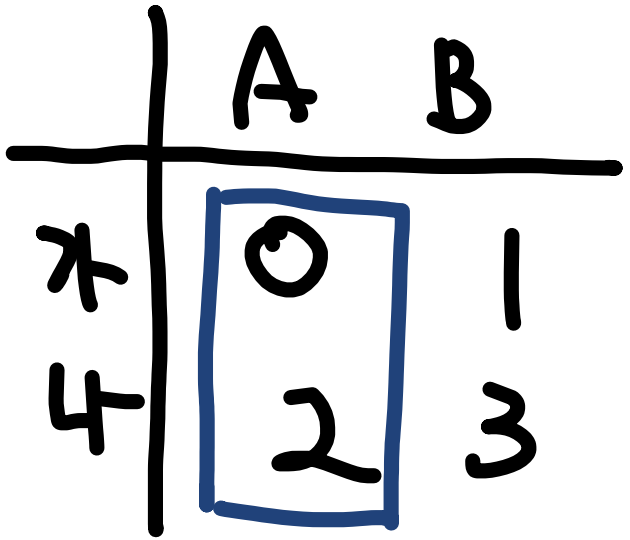

DATA
.CSV
.Xlsx
sql
pd.read_...()
DataFrame

NumPy

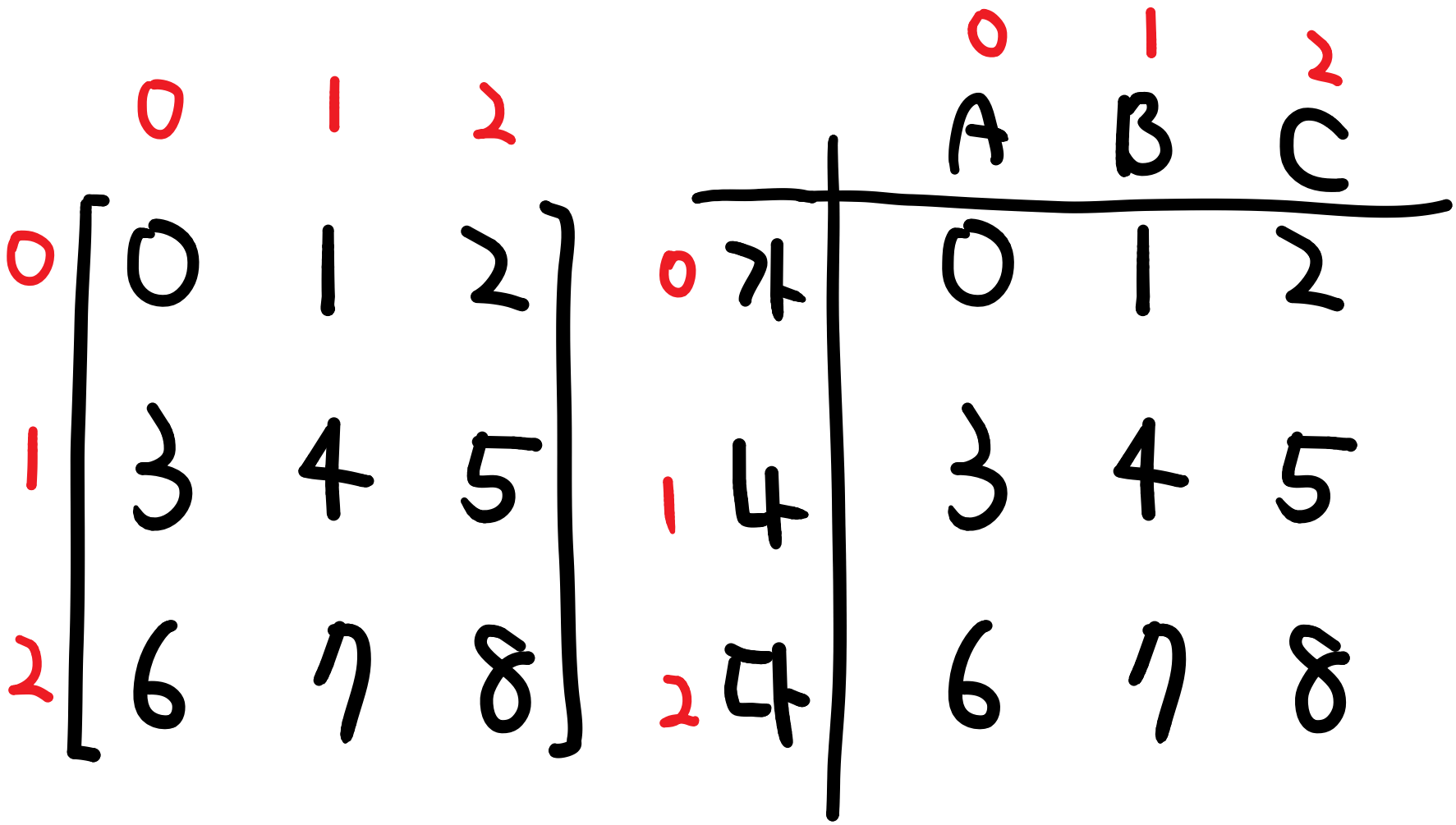


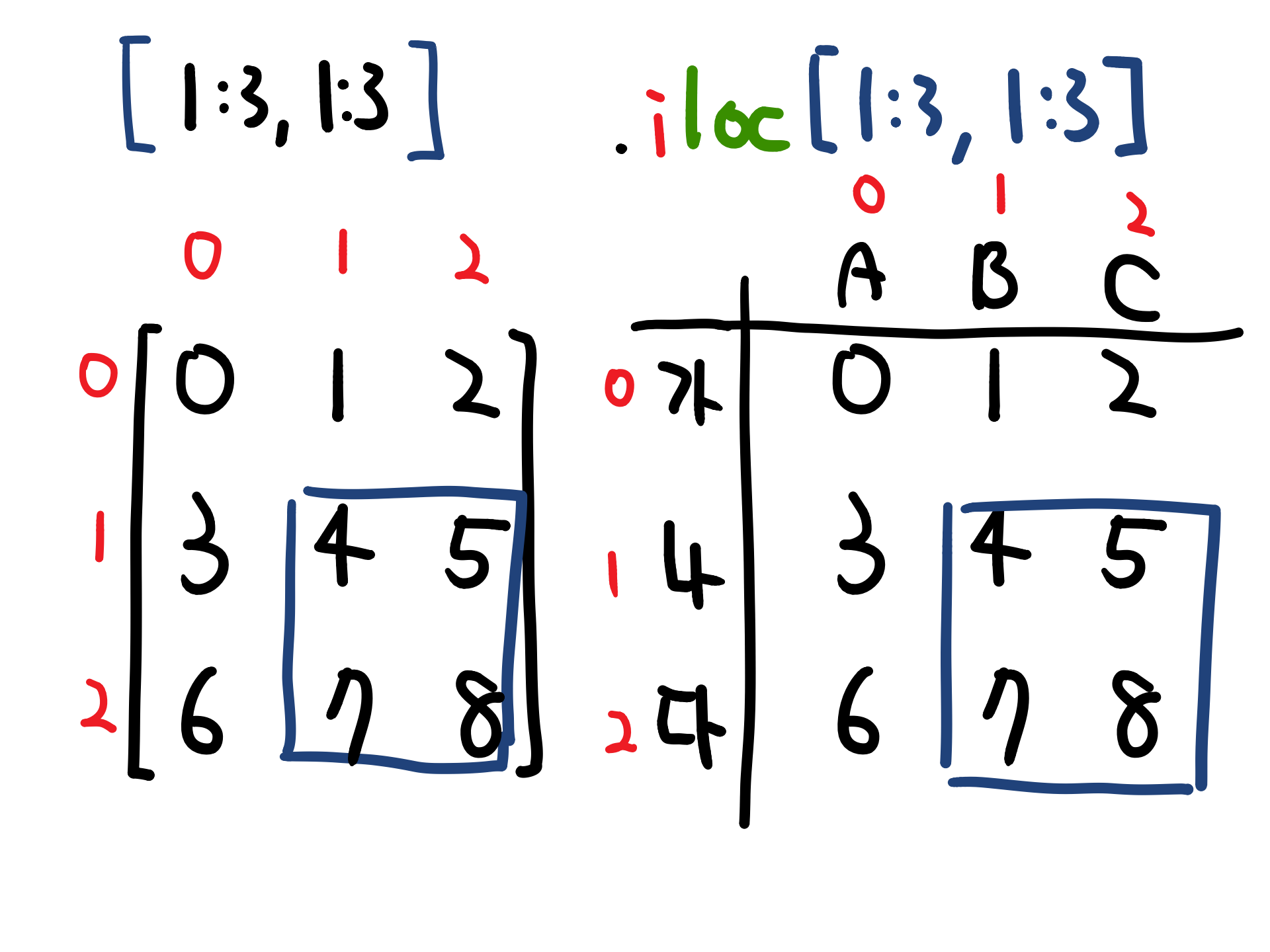
[0] → [0, 1]

Pandas



[A] → [0, 2]


0 1 2
0 1 2
3 4 5
6 7 8
A B C
0 1 2
0 1 2
3 4 5
6 7 8


[1:3, 1:3]
.iloc[1:3, 1:3]
0 1 2
0 1 2
0 1 2
3 4 5
6 7 8
0 1 2
A B C
0 1 2
3 4 5
6 7 8

pandas ← import pandas

VECTOR

0	A	X_1
1	B	X_2
2	C	X_3

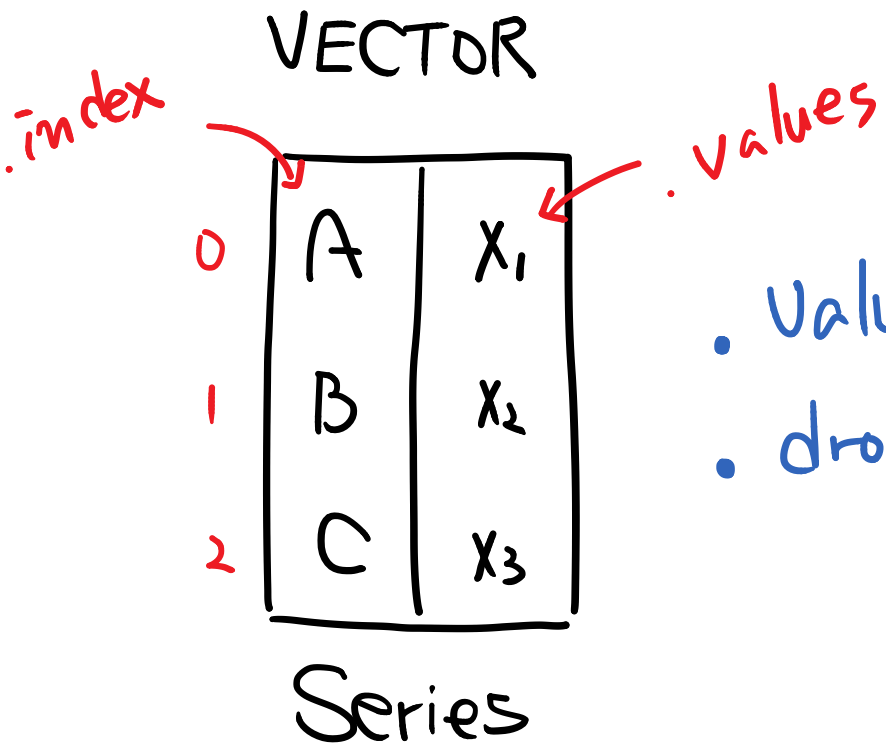
Series

MATRIX

		0	1
		A	B
0	가	1	2
1	나	3	4

DataFrame

pandas.Series



- .value_counts()
- .drop_duplicates()
- ⋮

pandas.DataFrame

MATRIX

	0	1
	A	B
0 가	1	2
1 나	3	4

DataFrame

지진 데이터 시각화, 선거와 이름통계 분석 등 실사례 사용

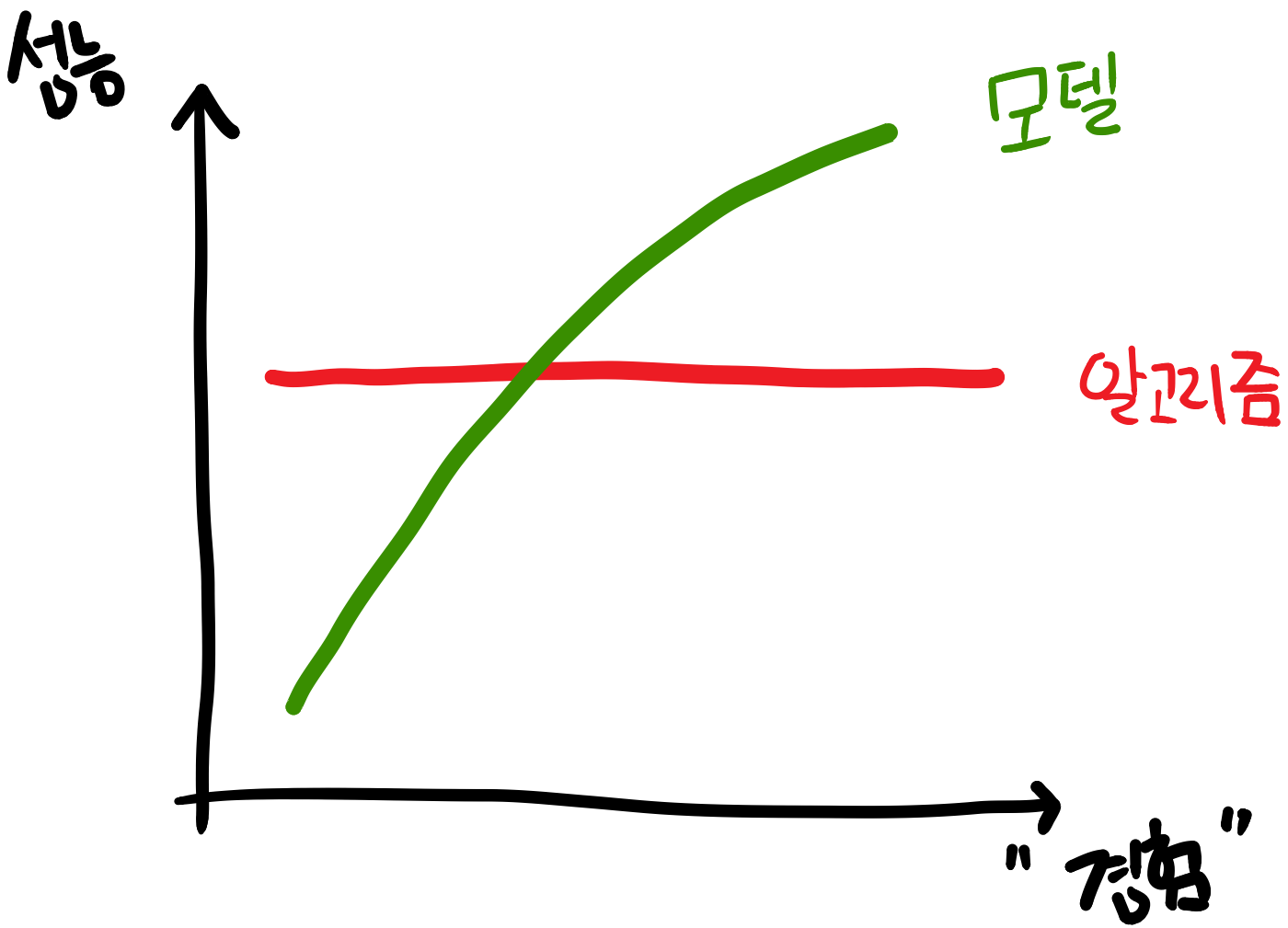
수정보완판
Python 3 / numpy 0.17.2

파이썬 라이브러리를 활용한 데이터 분석

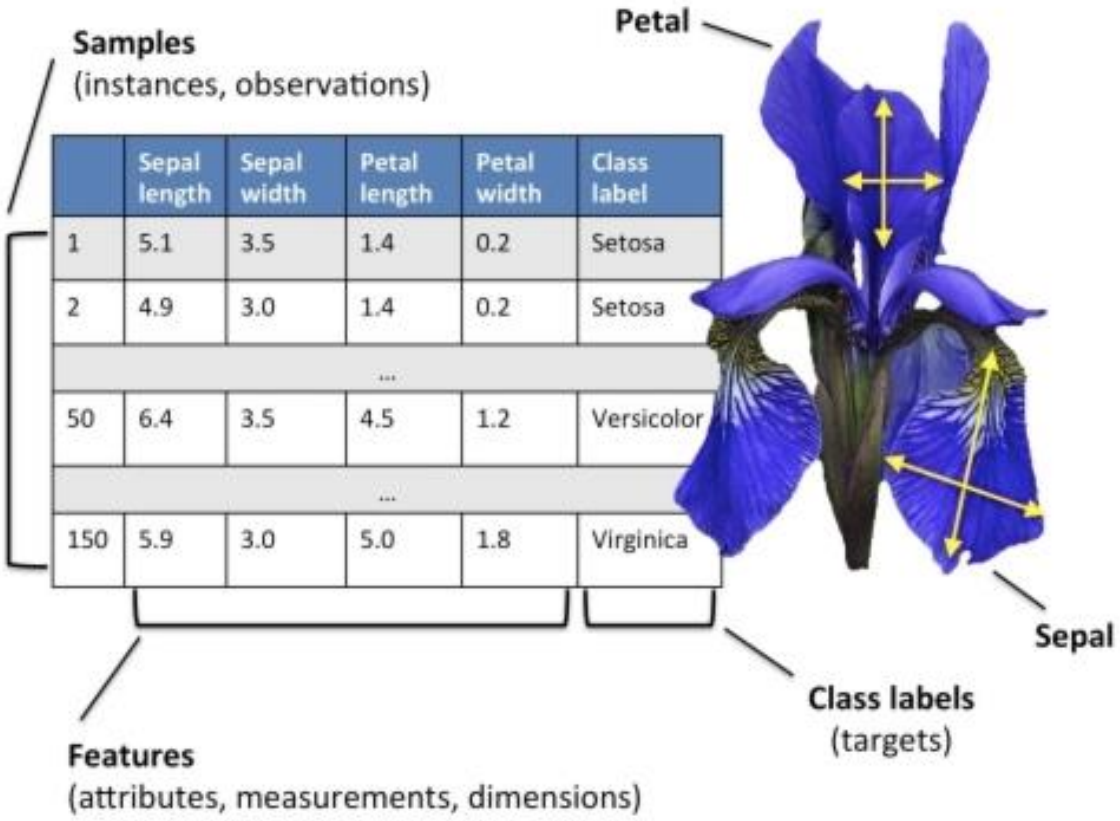
Python for Data Analysis

웨스 맥키니 지음
김영근 옮김

O'REILLY 한빛미디어

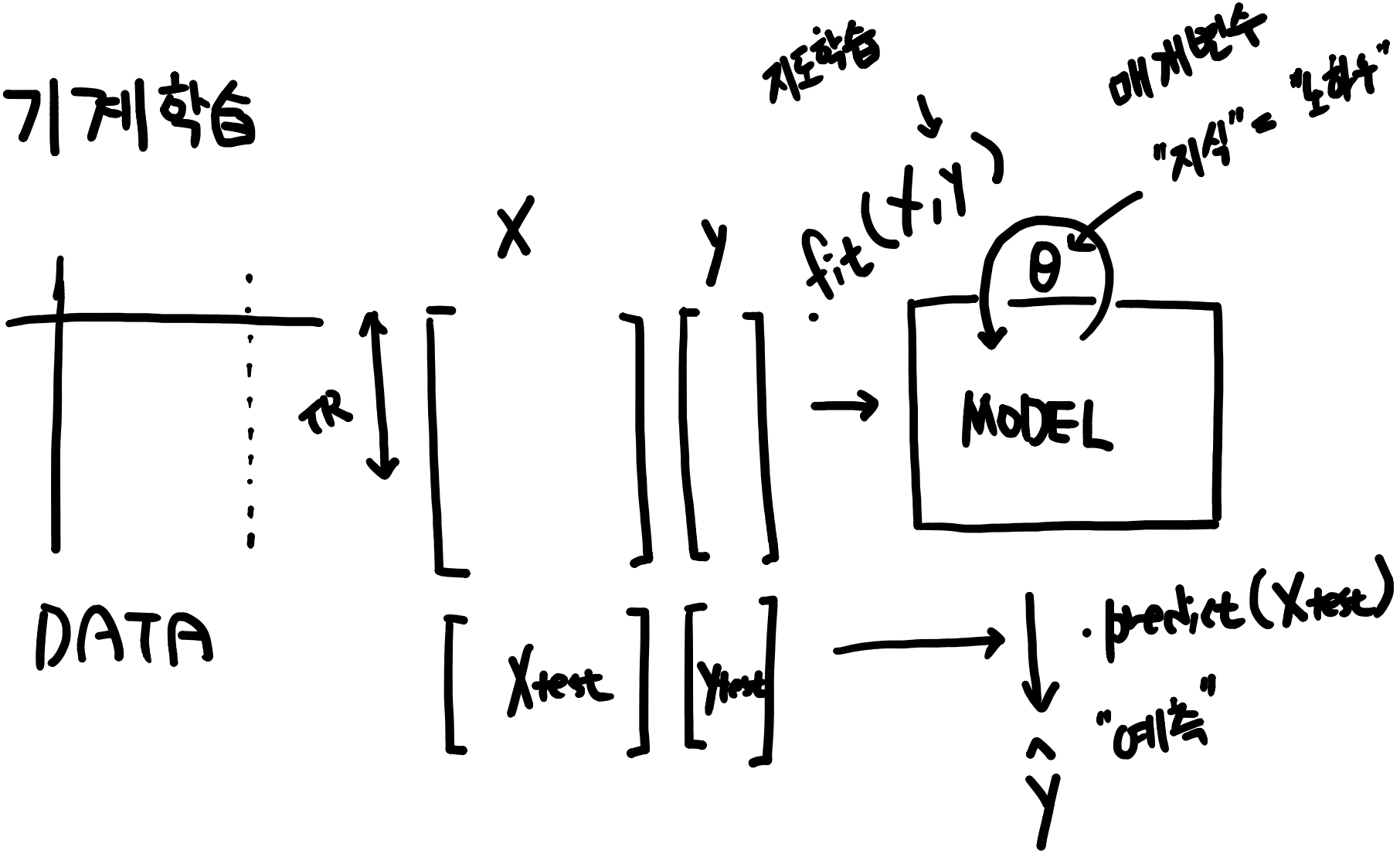

성능
모델
알고리즘
"정확"

샘플과 특징



	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

기계학습

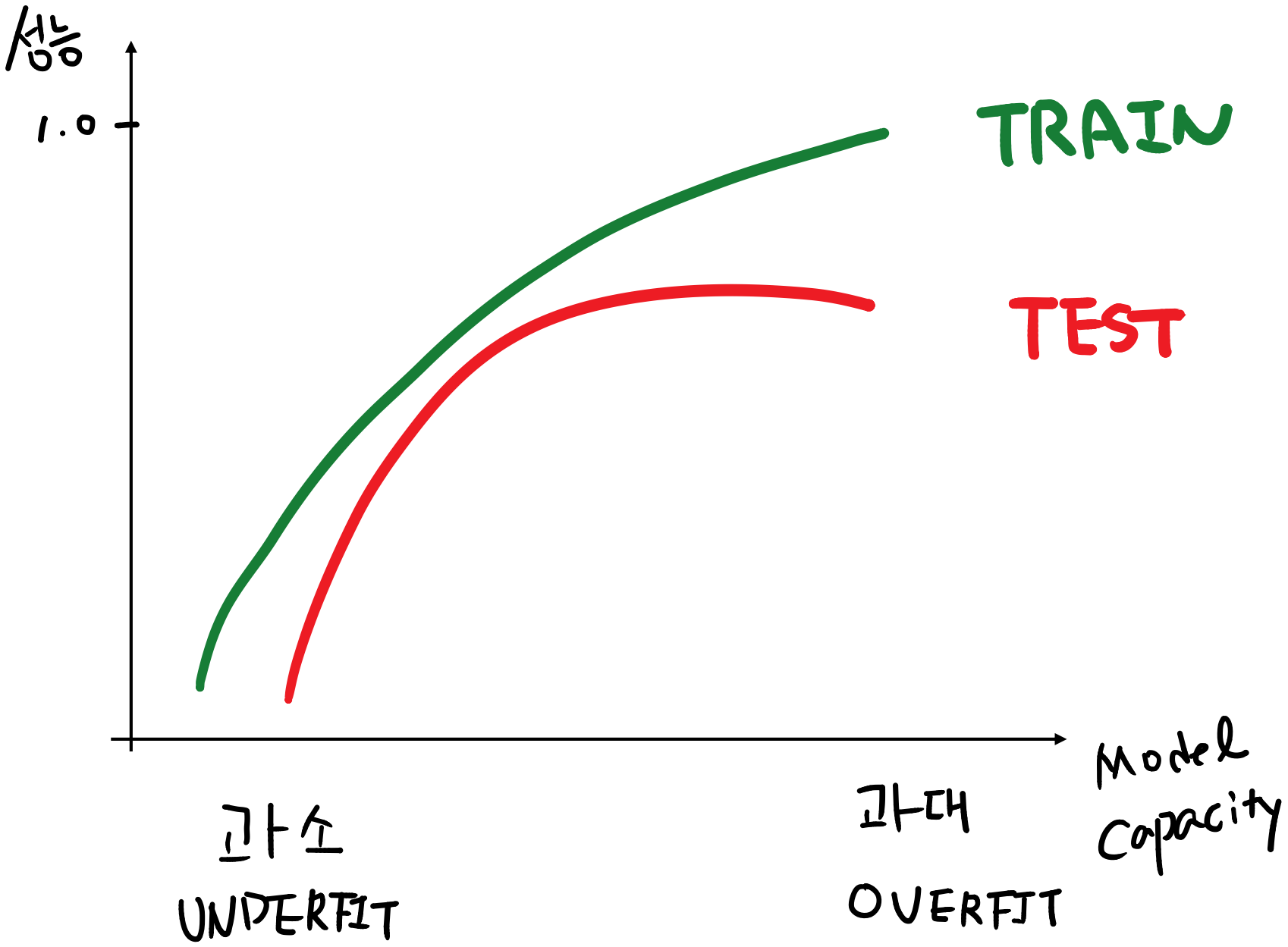

지도학습
매개변수
"지식" = "모형"
X
y
fit(X,y)
θ
MODEL
TR
DATA
[Xtest]
[ytest]
.predict(Xtest)
"예측"
ŷ

기계학습 모델 목표

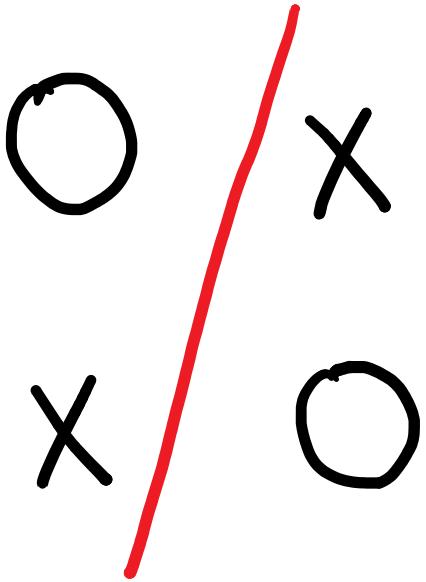
1. 최적화 ↔ 과소적합

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \, Loss(\theta)$$

2. 일반화 ↔ 과적합


성능
1.0
TRAIN
TEST
Model Capacity
과소
UNDERFIT
과대
OVERFIT

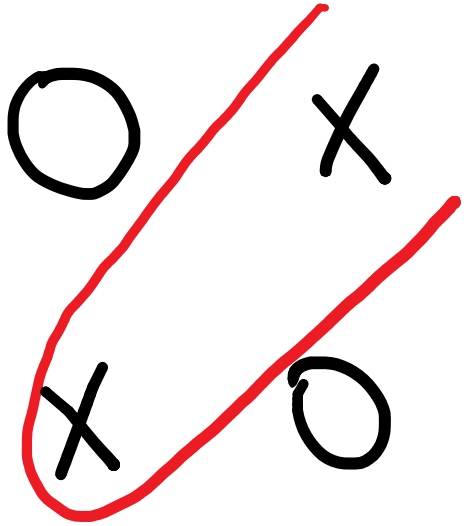
모델 표현력



"단순"

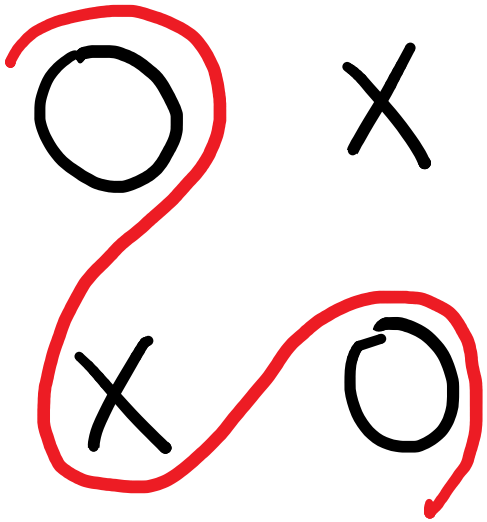
↓

과소적합



"적합"

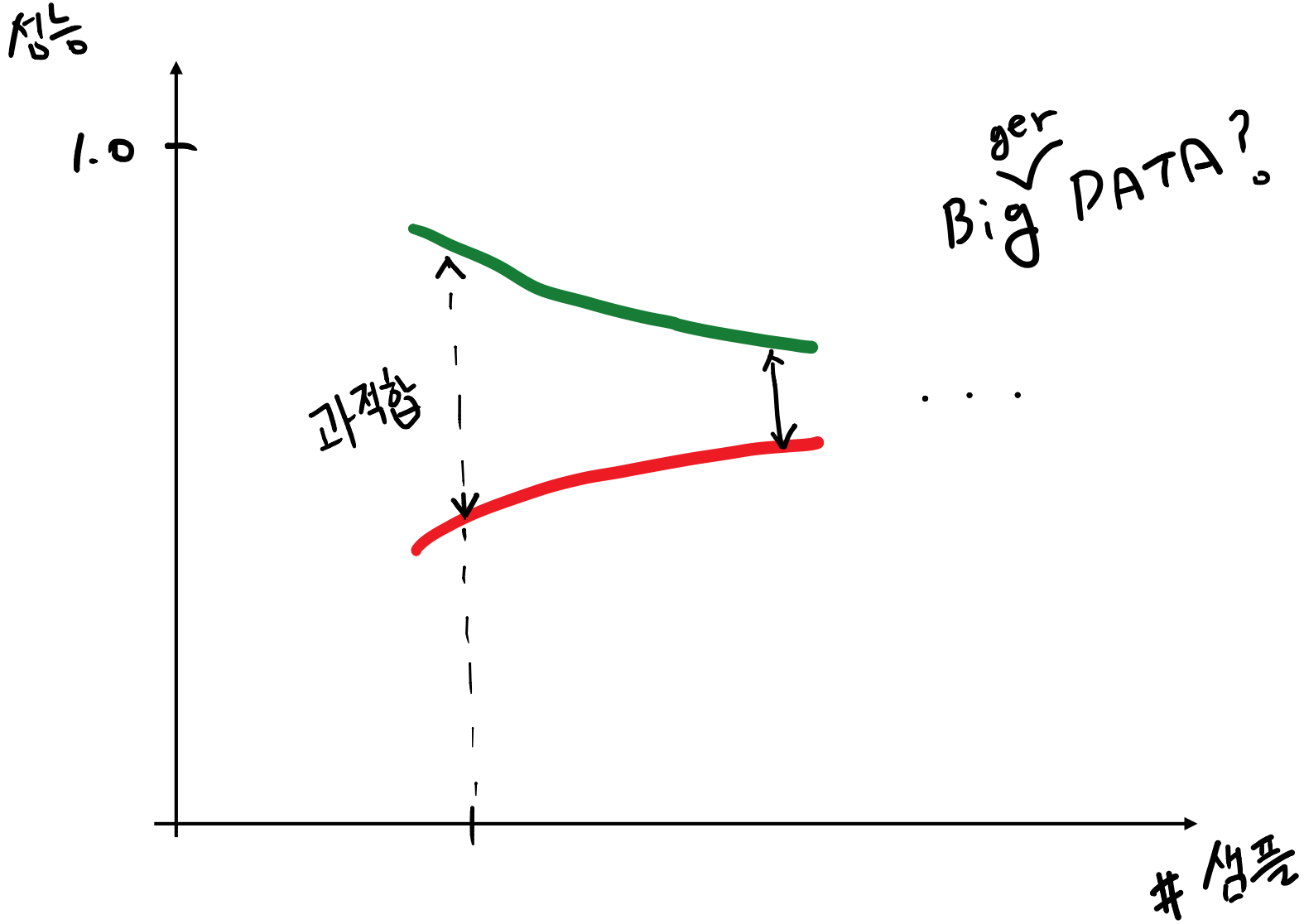
fit



"복잡"

↓

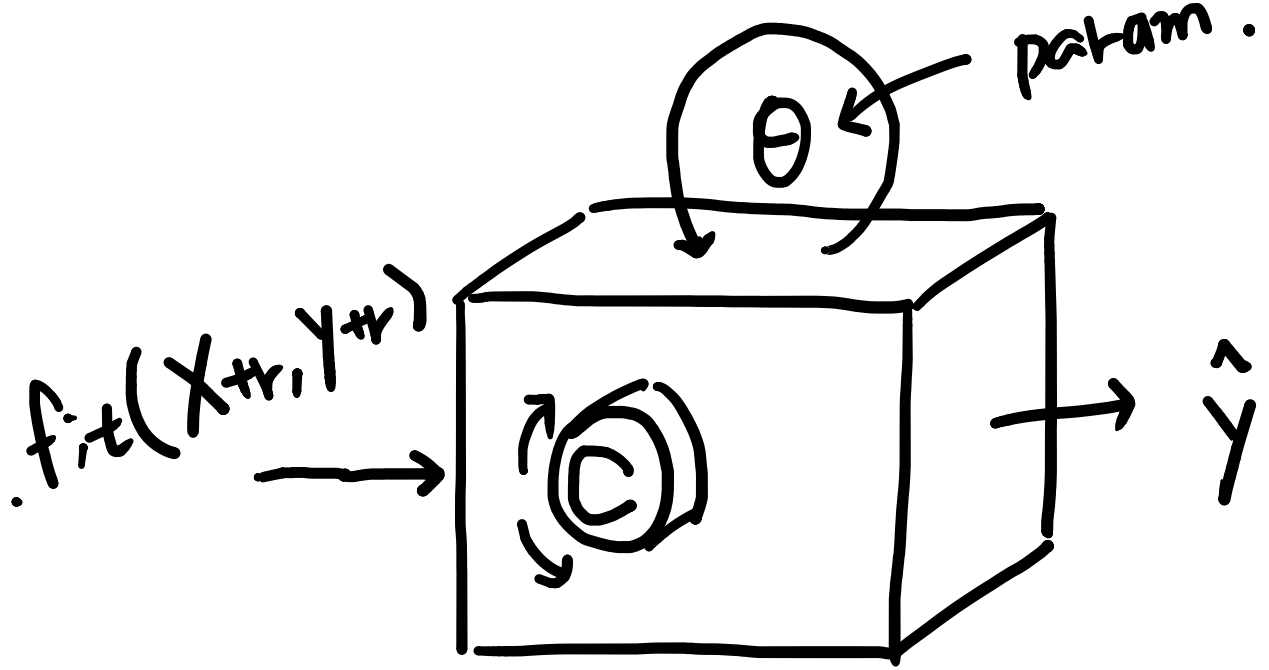
과적합



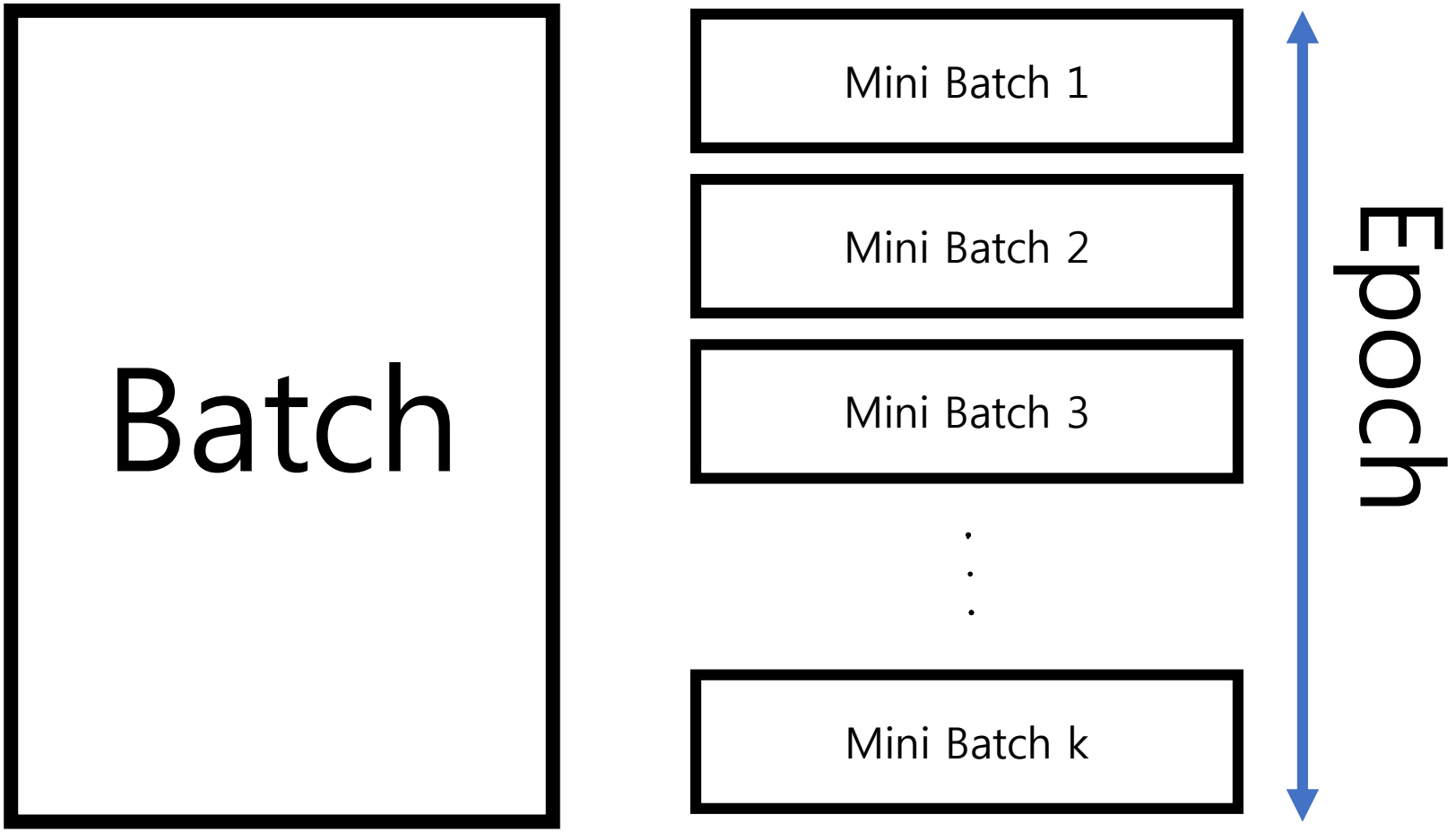
Bigger DATA?

. . .

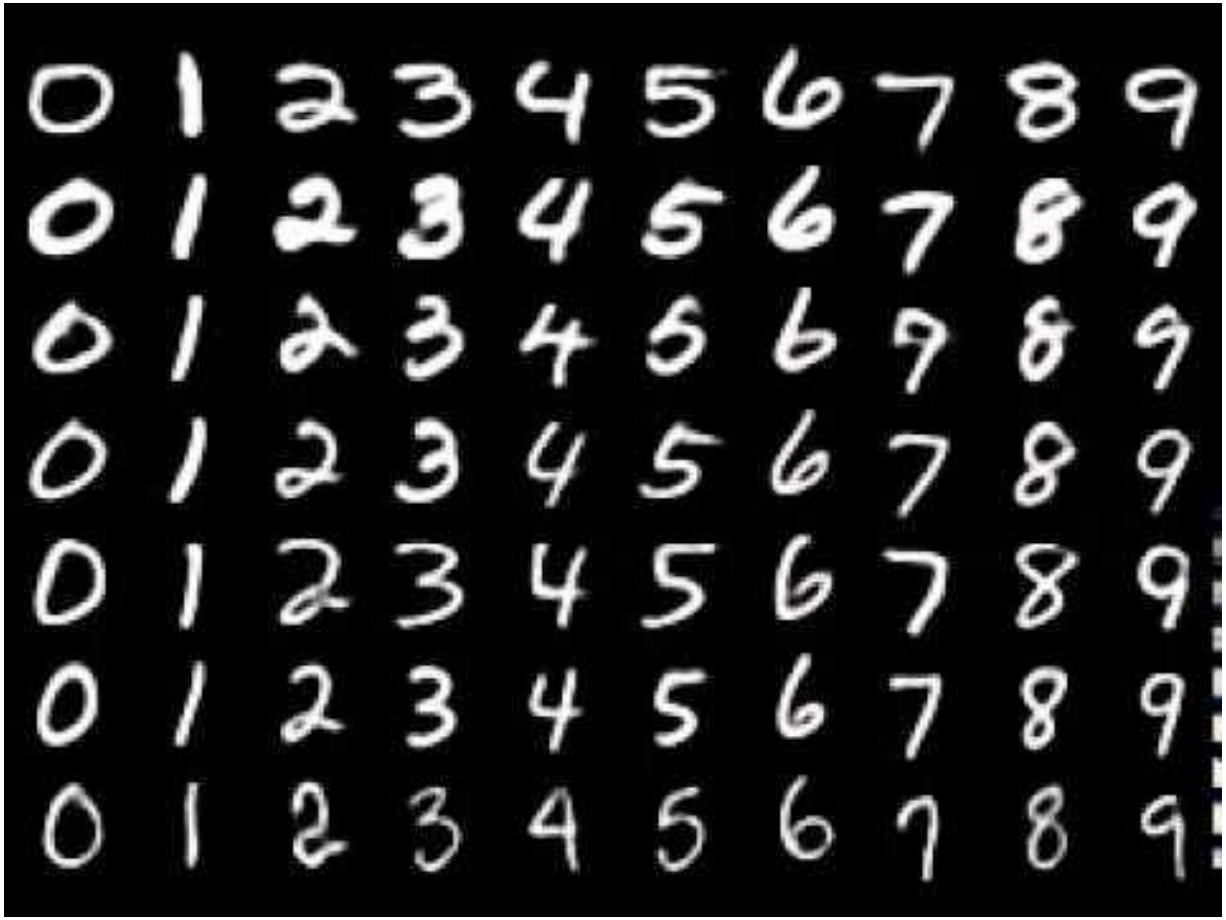
모델 튜닝 = Hyper-param. Tuning



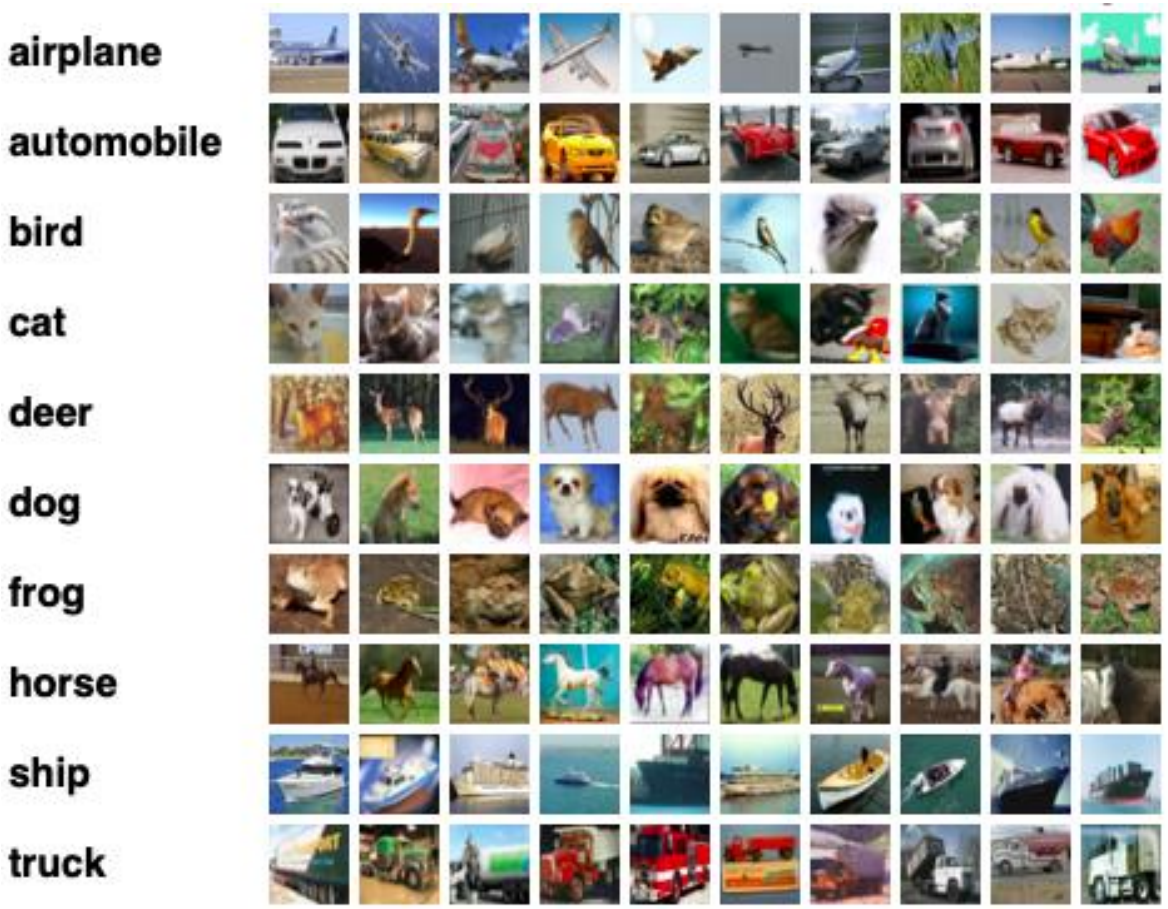
배치 (Batch)


Batch
Mini Batch 1
Mini Batch 2
Mini Batch 3
Mini Batch k
Epoch

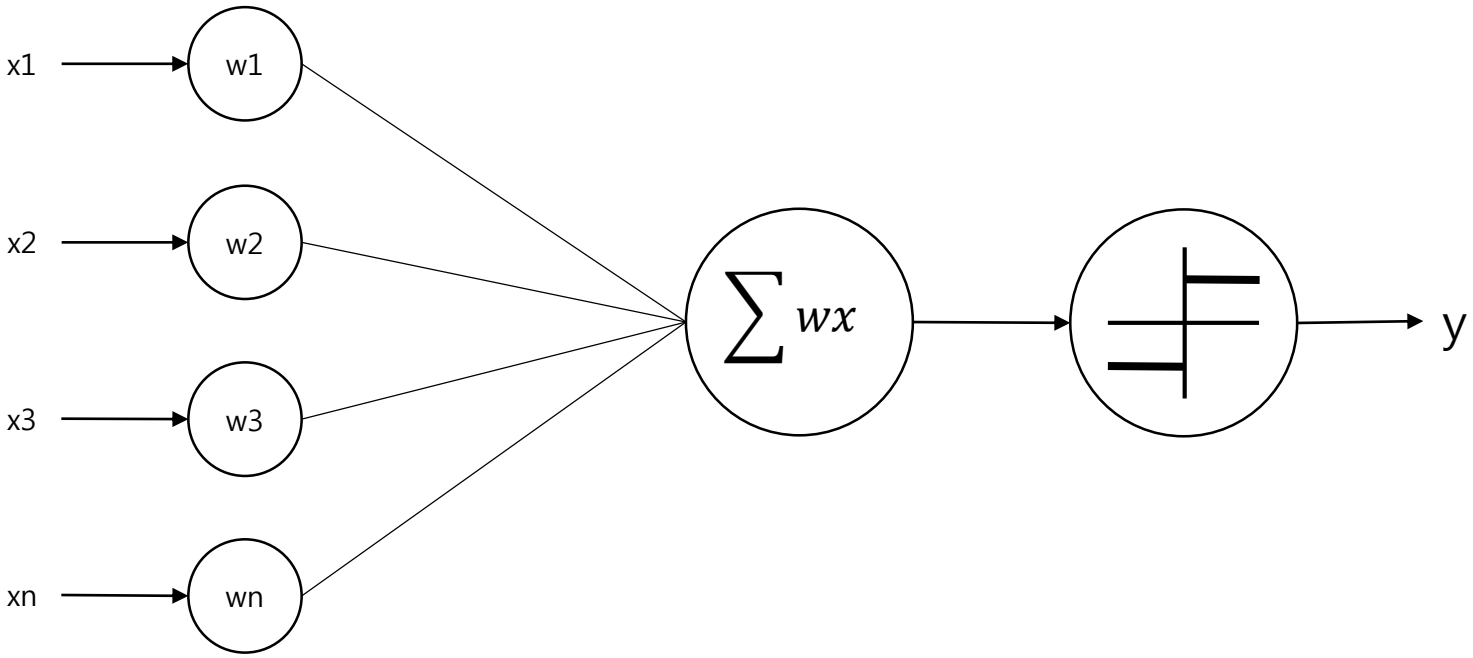
MNIST



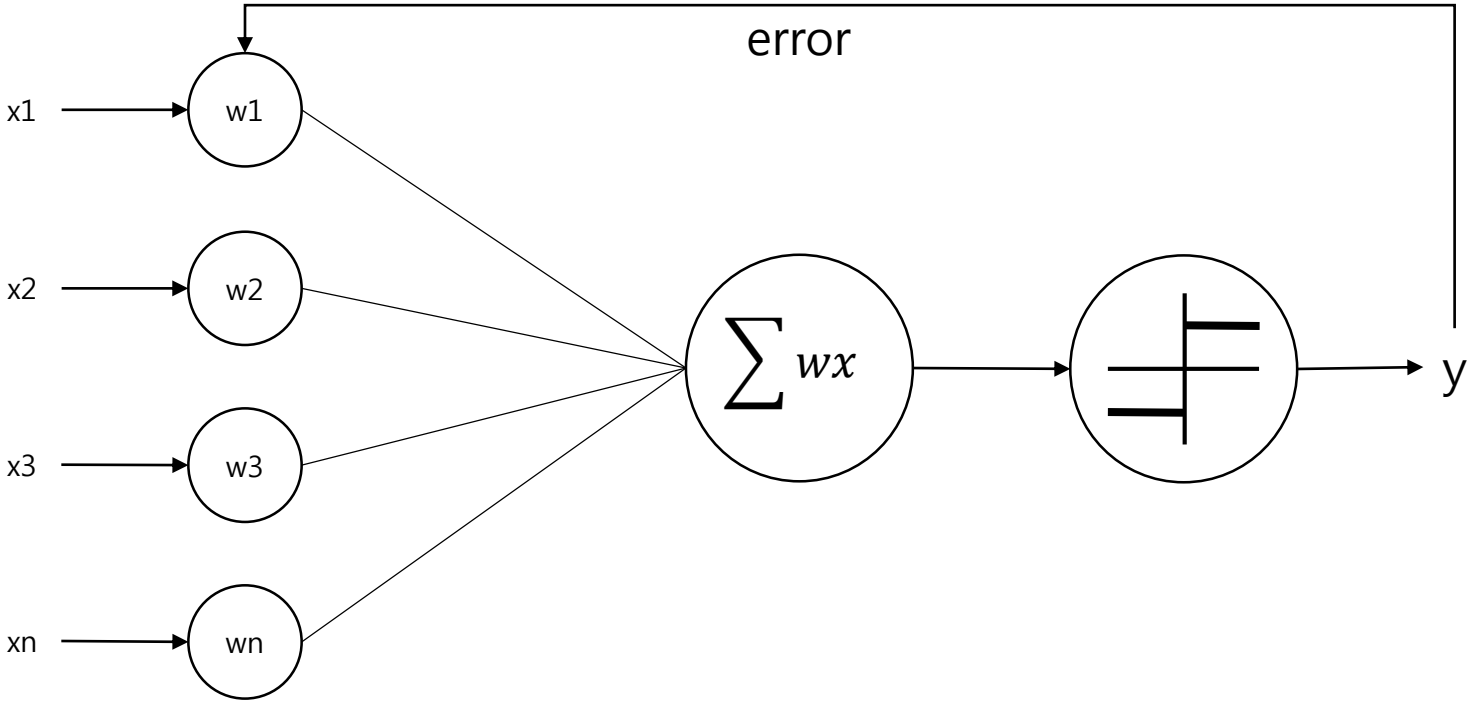
CIFAR


airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck

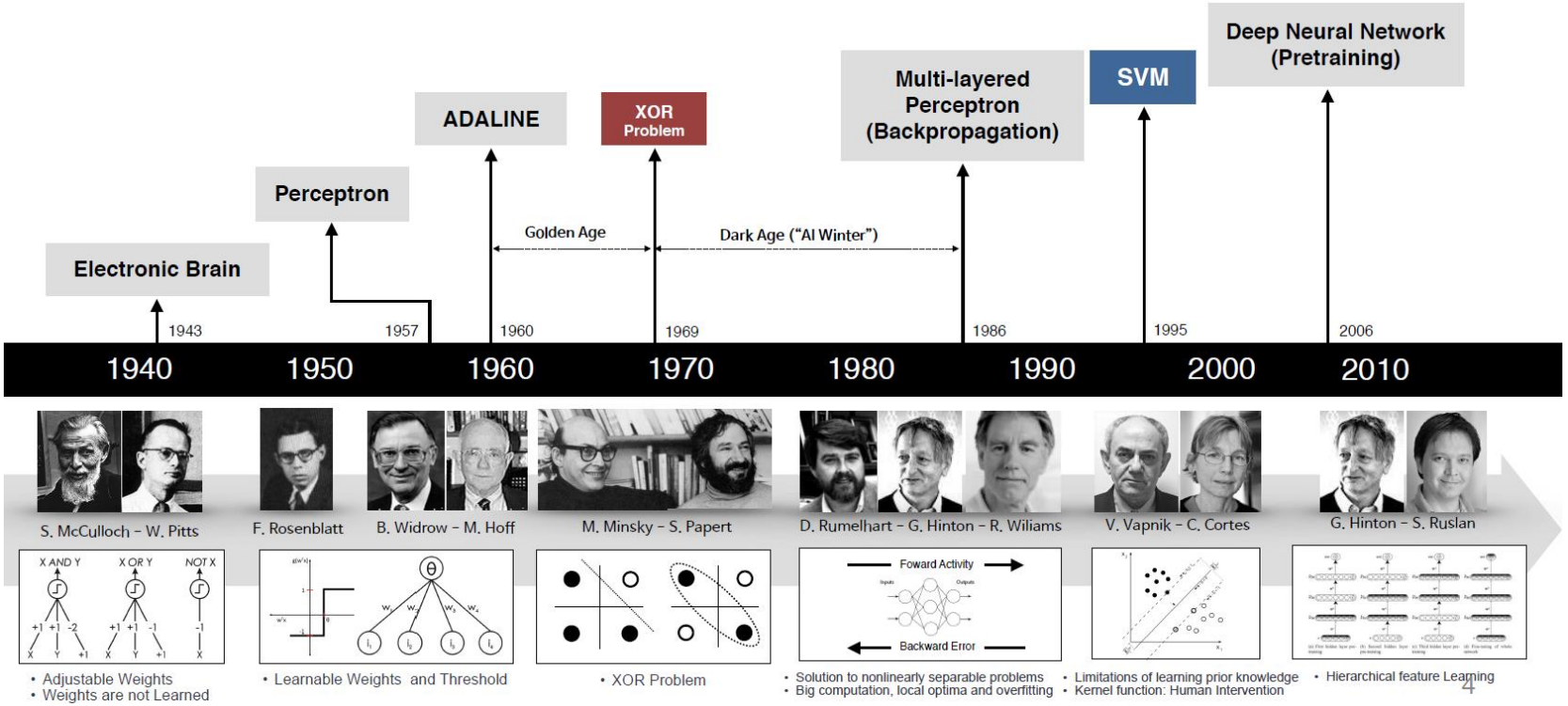
1943 McCulloch-Pitts 뉴런



1957 퍼셉트론

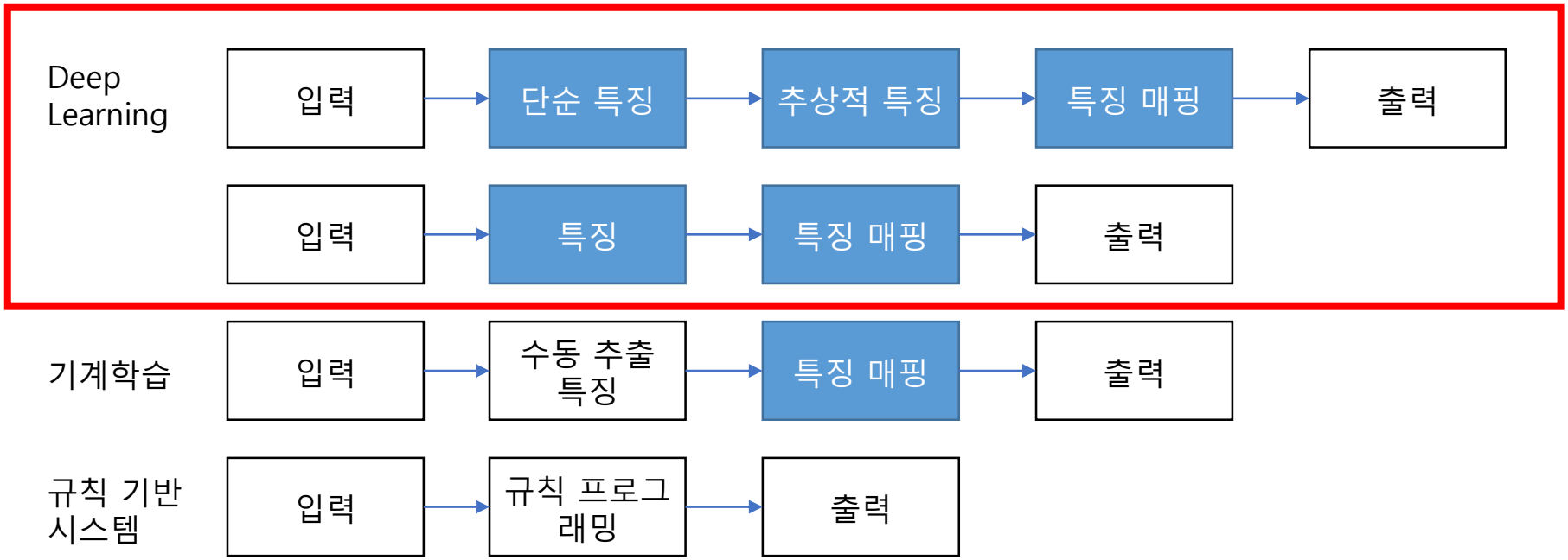


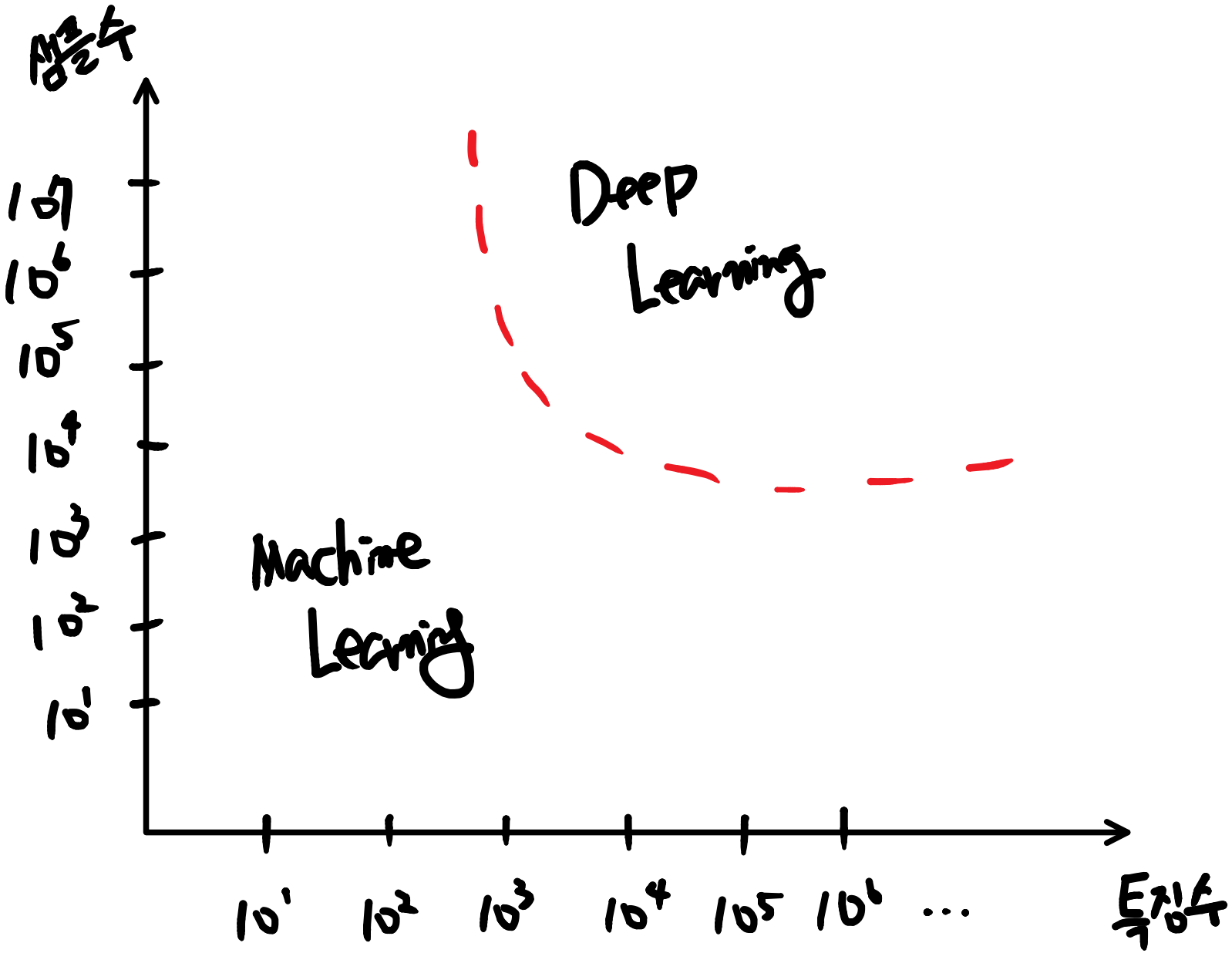
인공 신경망의 발전사


Deep Neural Network (Pretraining)
SVM
Multi-layered Perceptron (Backpropagation)
XOR Problem
ADALINE
Perceptron
Electronic Brain
Golden Age
Dark Age ("AI Winter")
1943
1957
1960
1969
1986
1995
2006
1940
1950
1960
1970
1980
1990
2000
2010
S. McCulloch - W. Pitts
F. Rosenblatt
B. Widrow - M. Hoff
M. Minsky - S. Papert
D. Rumelhart - G. Hinton - R. Wiliams
V. Vapnik - C. Cortes
G. Hinton - S. Ruslan
X AND Y
X OR Y
NOT X
Foward Activity
Backward Error
• Adjustable Weights
• Weights are not Learned
• Learnable Weights and Threshold
• XOR Problem
• Solution to nonlinearly separable problems
• Big computation, local optima and overfitting
• Limitations of learning prior knowledge
• Kernel function: Human Intervention
• Hierarchical feature Learning

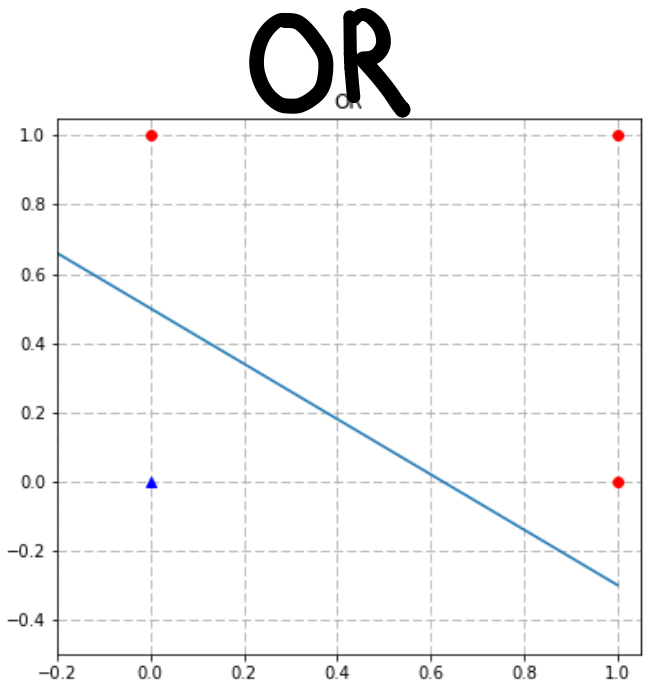
비교

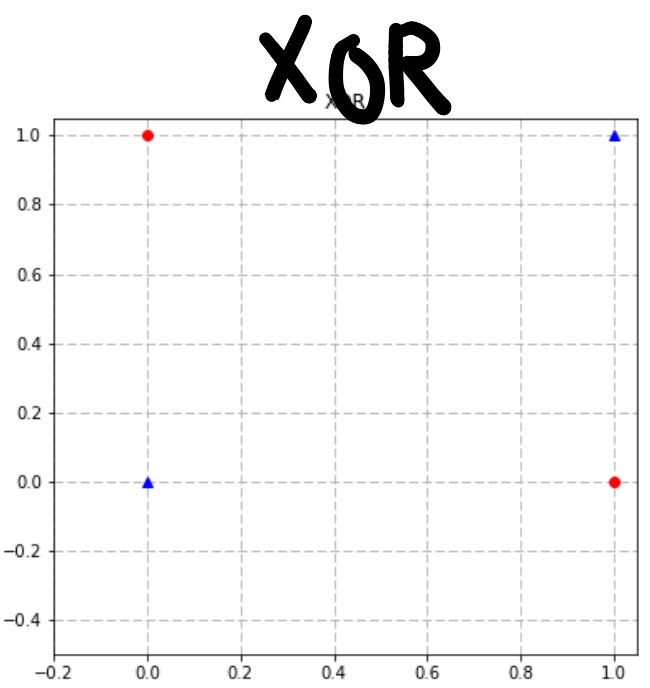
표현 기반 학습


Deep Learning
입력
단순 특징
추상적 특징
특징 매핑
출력
입력
특징
특징 매핑
출력
기계학습
입력
수동 추출 특징
특징 매핑
출력
규칙 기반 시스템
입력
규칙 프로그래밍
출력


샘플수
10^7
10^6
10^5
10^4
10^3
10^2
10^1
Deep Learning
Machine Learning
10^1 10^2 10^3 10^4 10^5 10^6 …
특징수

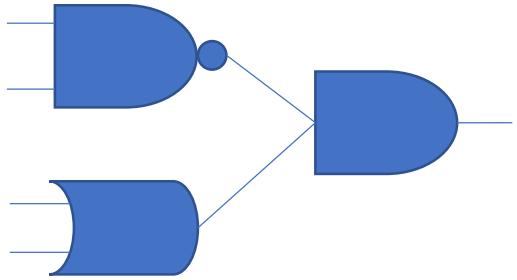
XOR 문제

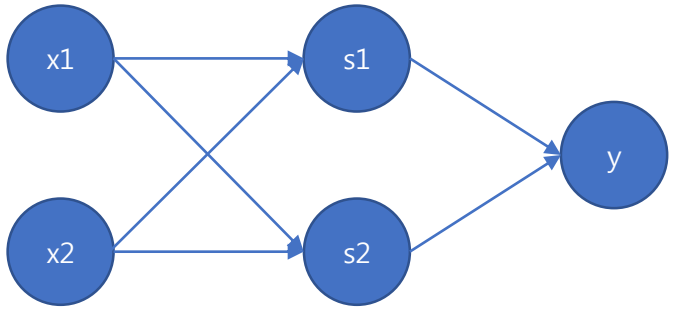

OR
OR
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
1.0


XOR
XOR
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
1.0

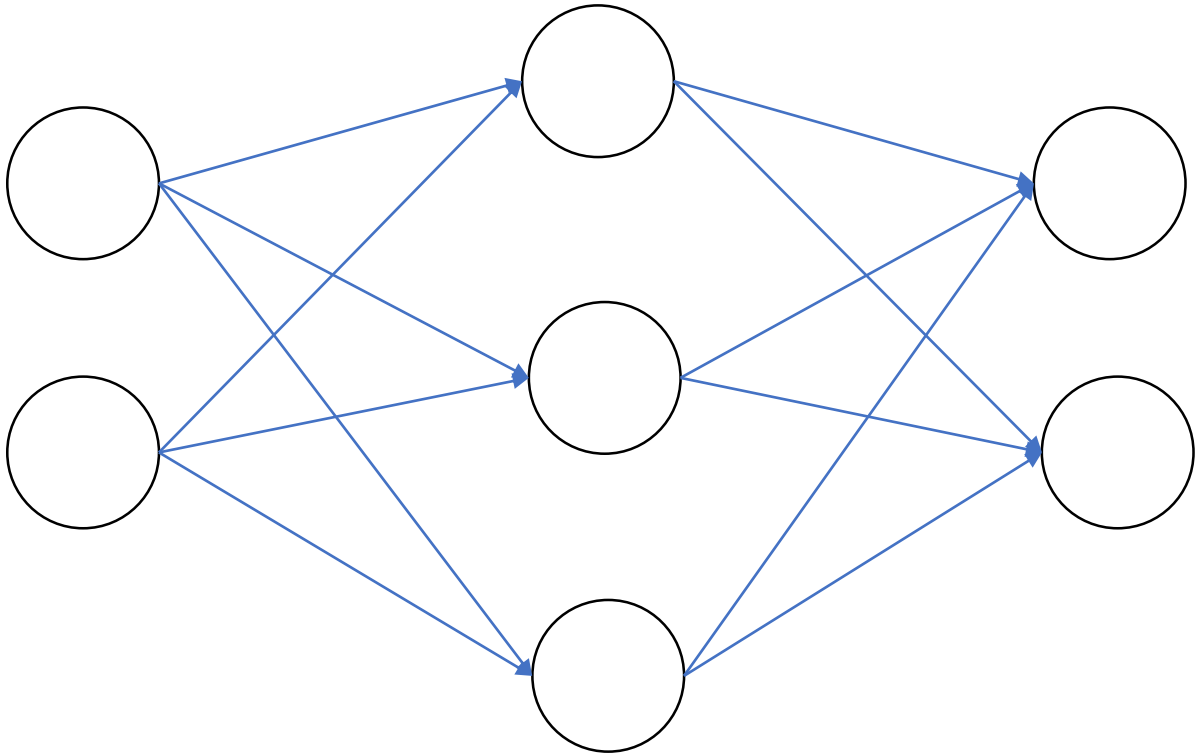
다층 퍼셉트론

x1	x2	s1	s2	y
0	0	1	0	0
1	0	1	1	1
0	1	1	1	1
1	1	0	1	0





신경망



손실 함수

회귀

- 평균 제곱 오차

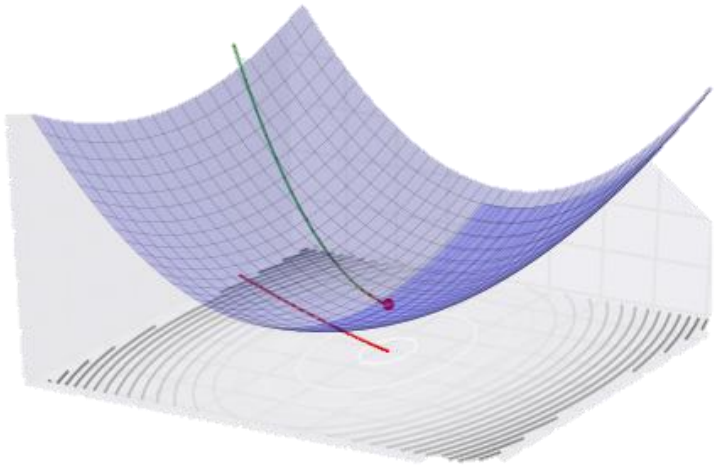
$$\frac{1}{2}\sum_{i}(y_i - \hat{y}_i)^2$$

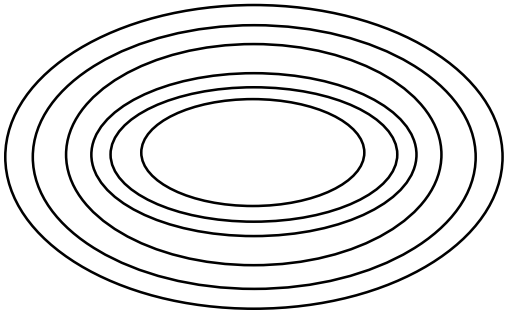
분류

- 교차 엔트로피 오류

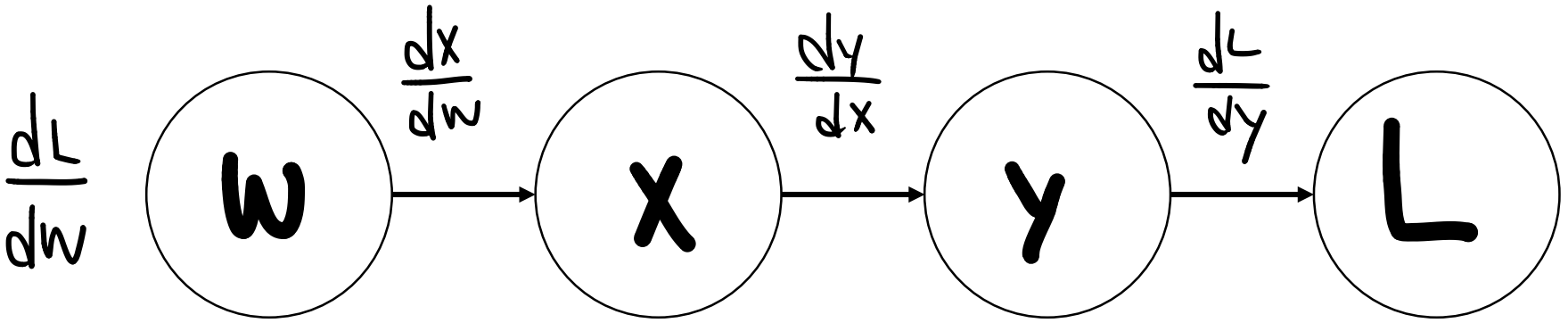
$$-\sum_{i} y_i \log \hat{y}_i$$

최적화





1986 오차역전파 (Back Propagation)

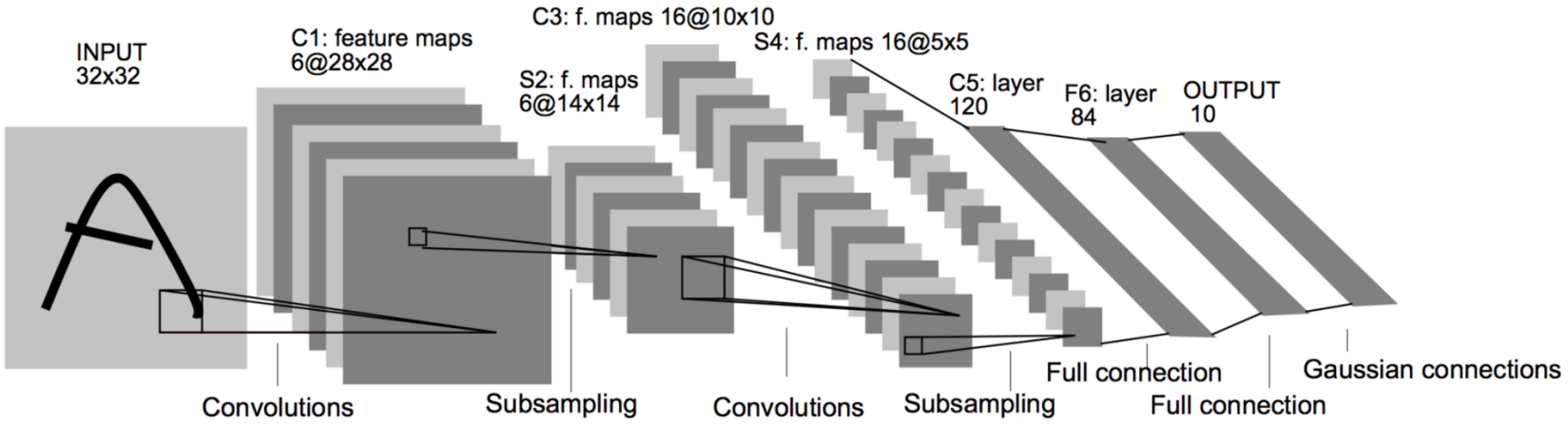


CNN

Convolutional Neural Network

합성곱 신경망

1997 LeNet


INPUT
32x32
C1: feature maps
6@28x28
S2: f. maps
6@14x14
C3: f. maps 16@10x10
S4: f. maps 16@5x5
C5: layer
120
F6: layer
84
OUTPUT
10
Convolutions
Subsampling
Convolutions
Subsampling
Full connection
Full connection
Gaussian connections

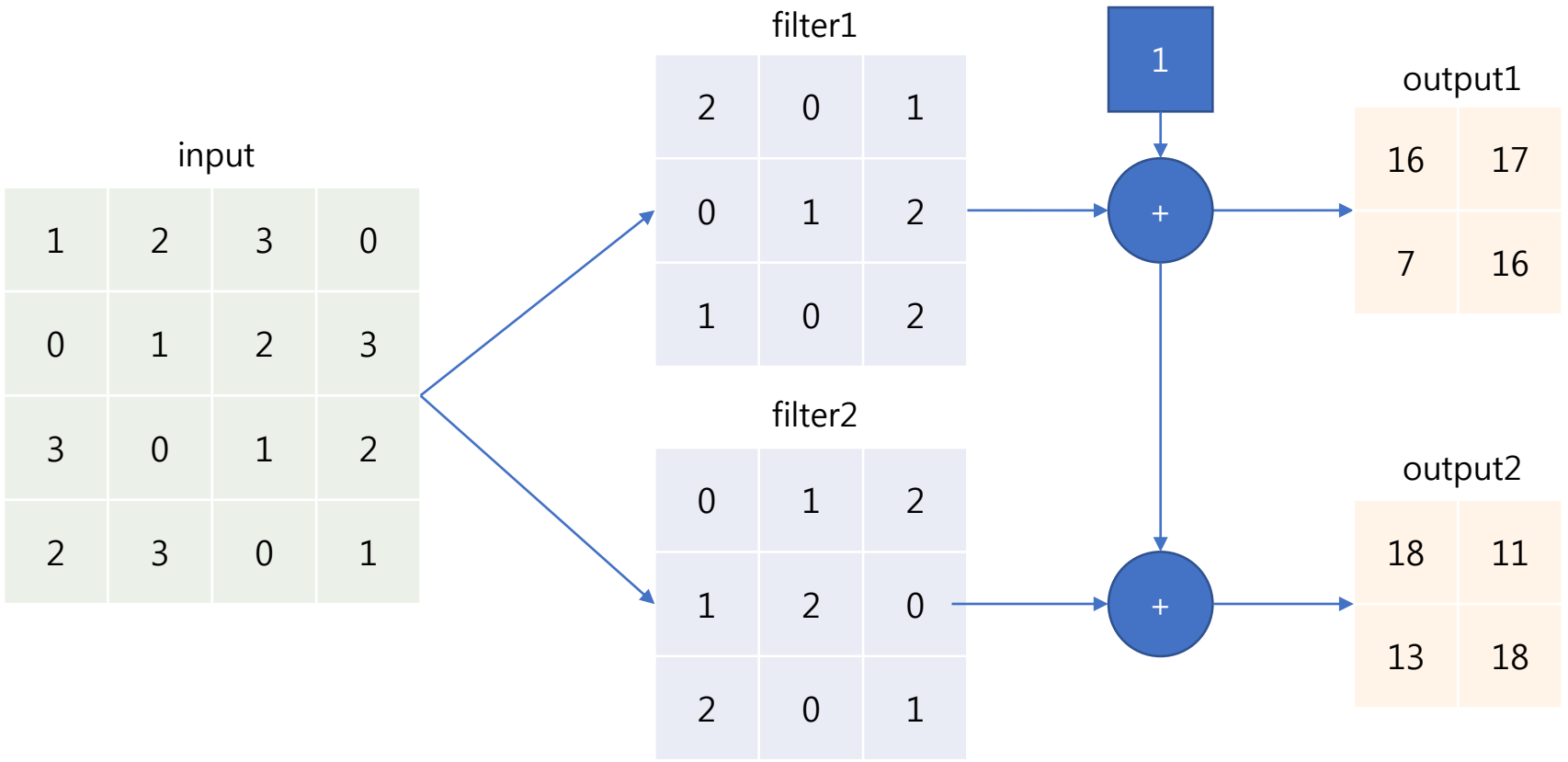
합성곱 연산

1	2	3	0
0	1	2	3
3	0	1	2
2	3	0	1

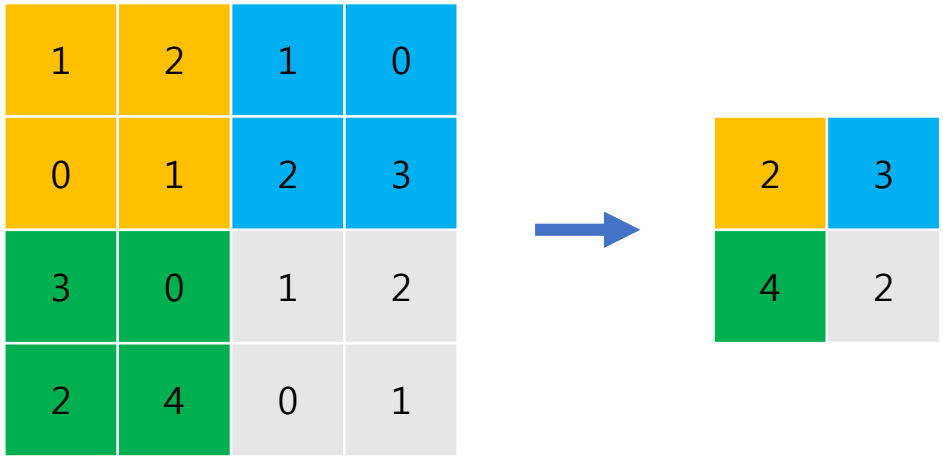
2	0	1
0	1	2
1	0	2

15	16
6	15

2D 합성곱 층


input
1 2 3 0
0 1 2 3
3 0 1 2
2 3 0 1
filter1
2 0 1
0 1 2
1 0 2
filter2
0 1 2
1 2 0
2 0 1
1
+
+
output1
16 17
7 16
output2
18 11
13 18

Max Pooling



RNN

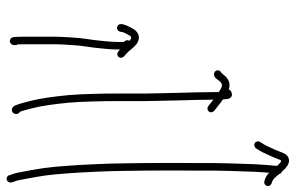
Recurrent Neural Network

매개변수 공유

Sequence

$x^{(1)}, \cdots, x^{(\tau)}$

RNN



$\theta = W, b$

Variable length input

Unfolding Graph

Recurrent

$$s^{(t)} = f(s^{(t-1)}; \theta)$$

params

$s^{(t)}$: system state

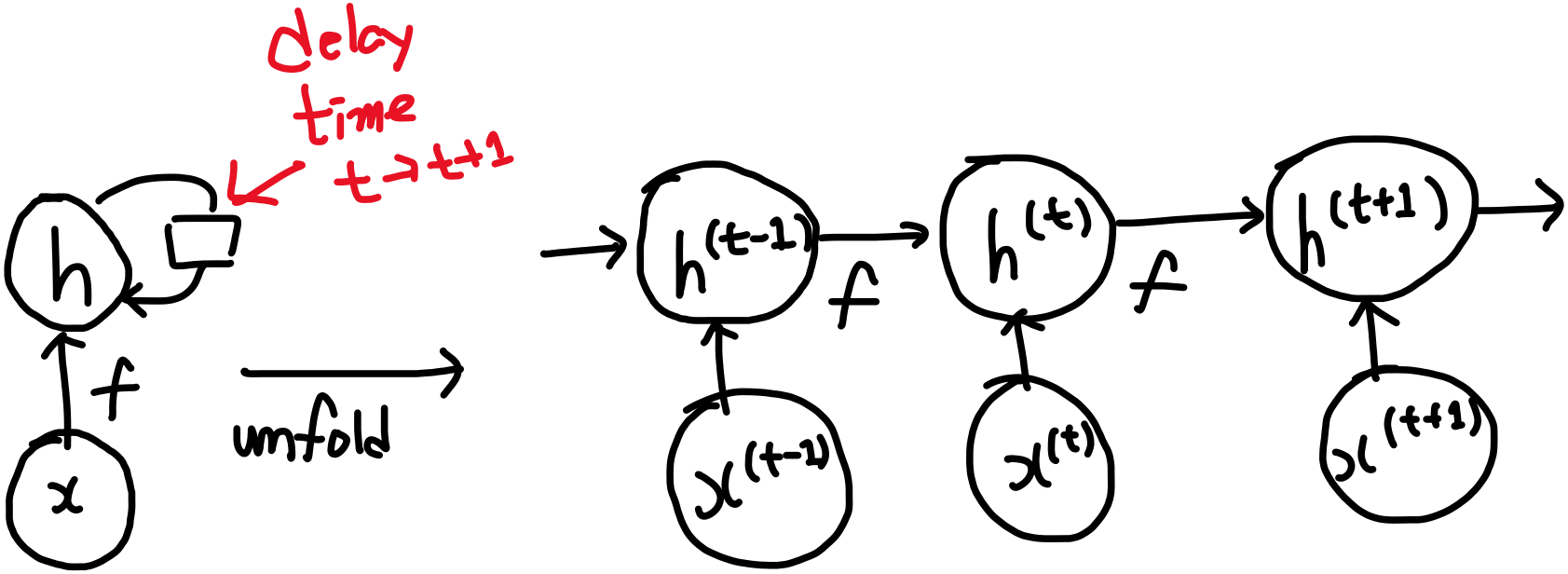
Unfold

$$s^{(3)} = f(s^{(2)}; \theta) = f(f(s^{(1)}; \theta); \theta)$$

$$s^{(t)} = f(s^{(t-1)}, x^{(t)}; \theta)$$

$x^{(t)}$: external signal

Recurrent Neural Network



Hidden state

EXTERNAL SIGNAL

$$h^{(t)} = f\left(h^{(t-1)}, x^{(t)}; \theta\right)$$

Unfolding과 매개변수 공유

$$h^{(t)} = g^{(t)}\left(x^{(t)}, x^{(t-1)}, x^{(t-2)}, \cdots, x^{(2)}, x^{(1)}\right)$$

"현재까지의 모든 입력"

$$= f\left(h^{(t-1)}, x^{(t)}; \theta\right)$$

θ ← W, b